41

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN

"ADMINISTRACION DE SISTEMAS ABIERTOS DE REDES DE COMPUTADORAS Y PRINCIPIOS BASICOS DE COMUNICACIONES"

T E S I S

QUE PARA OBTENER EL TITULO DE:

INGENIERO MECANICO ELECTRICISTA

P R E S E N T A N:

GLORIA PONCE VENEGAS

FRANCISCO IGNACIO CHAVEZ CASTAÑEDA

ASESOR: FIS. J. DE JESUS CRUZ GUZMAN

CUAUTITLAN IZCALLI, EDO. DE MEX. 1995

UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

UNIVERSIDAD NACIONAL AVENTMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

ASUNTO: VOTOS APROBATORIOS

DR. JAIME KELLER TORRES
DIRECTOR DE LA FES-CUAUTITLAN
P R E S E N T E .

AT'N: Ing. Rafael Rodriguez Caballos
Jefe del Departamento de Exámenes
Profesionales de la F.E.S. - C.

Con base en el art. 28 del Reglamento General de Exámenes, nos permitimos comunicar a usted que revisamos la TESIS TITULADA:

"Administración de sistemas abiertos de redes de computadoras y principios básicos de comunicaciones".

que presenta la pasante: Gloria Ponce Venegas

con número de cuenta: 8509571-1 para obtener el TITULO de:

Ingeniera Mecánica Electricista .

Considerando que dicha tesis reúne los requisitos necesarios para ser discutida en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

A T E N T A M E N T E .
"POR MI RAZA HABLARA EL ESPIRITU"
Cuautitlán Izcalli, Edo. de Méx., a 14 de agosto de 1995

PRESIDENTE	Fís. J. de Jesús Cruz Guzmán	
VOCAL	Ing. José Luis Rivera López	
SECRETARIO	Ing. Ubaldo Ramírez Urizar	
PRIMER SUPLENTE	Ing. Nicolás Calva Tapia	
SEGUNDO SUPLENTE	Ing. Rogelio Ramos Carranza	

UAE/DEP/VAP/02

UNIVERSIDAD NACIONAL AVENMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

ASUNTO: VOTOS APROBATORIOS

DR. JAIME KELLER TORRES
DIRECTOR DE LA FES-CUAUTITLAN
P R E S E N T E .

ATN: Ing. Rafael Rodriguez Ceballos
Jefe del Departamento de Exámenes
Profesionales de la F.E.S. - C.

Con base en el art. 28 del Reglamento General de Exámenes, nos permitimos comunicar a usted que revisamos la TESIS TITULADA:

"Administración de sistemas abiertos de redes de computadoras y principios básicos de comunicaciones"

que presenta el pasante: Francisco Ignacio Chávez Castañeda

con número de cuenta: 8300319-2 para obtener el TITULO de:

Ingeniero Mecánico Electricista .

Considerando que dicha tesis reúne los requisitos necesarios para ser discutida en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

A T E N T A M E N T E .
"POR MI RAZA HABLARA EL ESPIRITU"
Cuautitlan Izcalli, Edo. de Méx., a 14 de agosto de 1995

PRESIDENTE	Fis. J. de Jesús Cruz Guzmán	
VOCAL	Ing. José Luis Rivera López	
SECRETARIO	Ing. Ubaldo Ramírez Urizar	
PRIMER SUPLENTE	Ing. Nicolás Calva Tapia	
SEGUNDO SUPLENTE	Ing. Rogelio Ramos Carranza	

UAE/DEP/VAP/02

Índice

Introducción

Capítulo I

Conceptos Generales	1
1.1 Concepto de RED	3
1.2 Objetivos de las redes	6
1.2.1 Compartir recursos	7
1.2.2 Mejorar la confiabilidad de los sistemas	7
1.2.3 Reducir Costos	7
1.2.4 Como medio de comunicación	8
1.3 Clasificación de las redes	8
1.4 Tipos de redes	10
1.4.1 Red punto a punto	10
1.4.2 Red de transmisión por canal	11
1.5 Multiplexado	12
1.5.1 Multiplexado por división de frecuencia	12
1.5.2 Multiplexado por división de tiempo	13
1.6 Tipos de redes de conmutación	15
1.6.1 Circuitos de conmutación	15
1.6.2 Conmutación de paquetes	17
1.6.3 Circuitos virtuales y datagramas	18
1.7 Topologías	20
1.7.1 Control	20
1.7.2 Modelos de topologías	22
1.8 Control de Acceso al Medio (MAC)	30
1.8.1 Contención	30
1.8.2 Probar y seguir	32
1.9 Arquitectura Cliente/Despachador	34
1.10 Sistema Abierto	35

Capítulo II

Principios de Telecomunicaciones	37
2.1 Codificación de la información	39
2.2 Formas de transmisión	42
2.3 Sistema básico de comunicación	44
2.4 Enlaces de comunicación	47
2.5 Frecuencia, espectro y ancho de banda	48
2.6 Relación entre velocidad de transmisión y ancho de banda	53
2.7 Intensidad de la señal	60
2.8 Transmisión digital y transmisión analógica	61

2.9 Formatos binarios de señales digitales 63
2.9.1 Señal sin retorno a cero (NRZ) 64
2.9.2 Señal sin retorno a cero bipolar (NRZB) 64
2.9.3 Señal con retorno a cero (RZ) 65
2.9.4 Señal con retorno a cero bipolar (RZB) 66
2.9.5 Codificación Manchester 66
2.9.6 Codificación Manchester diferencial 67
2.9.7 Señal sin retorno a cero con inversión de marco 67
2.10 Adecuación de la información 68
2.11 Medios de transmisión 69
2.11.1 Par trenzado 70
2.11.2 Cable coaxial 72
2.11.3 Fibra óptica 74

Capítulo III

Normas y arquitecturas de redes 78
3.1 Normas para redes 80
3.1.1 Modelo de referencia OSI 80
3.1.2 Estándar 802 del IEEE 85
3.1.3 Interfase de Datos Distribuidos por Fibra Óptica (FDDI) 93
3.2 Interconexión de redes 96
3.2.1 Repetidores 98
3.2.2 Puentes (Bridges) 99
3.2.3 Ruteadores (Routers) 100
3.2.4 Pasarelas (Gateway) 102
3.3 Protocolos 103
3.3.1 Protocolo de interconexión de redes (IP) 105
3.3.2 Protocolo de control de transmisión (TCP) 109
3.4 Protocolos superiores 113
3.4.1 Protocolo Telnet 113
3.4.2 Protocolo FTP 114
3.4.3 Protocolo SNMP 115

Capítulo IV

Administración de Redes 119
4.1 Administración de redes 122
4.2 Marco de trabajo para la administración de redes OSI 124
4.2.1 Administración de configuración de la red 125
4.2.2 Administración de fallas 125
4.2.3 Administración de rendimiento 126
4.2.4 Administración de seguridad 127
4.2.5 Administración de costo de uso de la red 127

4.2.6 Software para la administración de la red 128
4.3 Administración de una máquina Unix 130
4.3.1 Administración del acceso a una máquina Unix 133
4.3.2 El sistema de archivos visto por el administrador 136
4.3.3 Encendido y apagado de una máquina Unix 137
4.3.4 Montar y desmontar sistemas de archivos 140
4.3.5 Respaldos 144
4.3.6 Periféricos 147
4.3.7 Acceso y manejo de redes 150
4.4 Seguridad en un sistema Unix 153
4.4.1 Problemas con las contraseñas (passwords) 155
4.4.2 Problemas con el identificador del usuario (User ID) 157
4.4.3 Revisión del archivo /usr/spool/cron/crontab 159
4.4.4 Permisos de archivos importantes 161
4.4.5 Problemas con las terminales inteligentes 162

Capítulo V

Seguridad 164
5.1 Seguridad en sistemas de cómputo 167
5.2 Políticas de seguridad 169
5.3 Formas de ataque mejor conocidas 173
5.3.1 El gusano de Internet 174
5.3.2 El intruso astuto 176
5.3.3 El caballo de Troya 178
5.4 Virus en Unix 182
5.5 Inspección de paquetes 185

Capítulo VI

Formas de defensa 187
6.1 Medidas preventivas 188
6.2 Seguridad en contraseñas 188
6.3 Auditoría de sistemas 192
6.4 Identificación y cifrado 196
6.4.1 El proyecto Athena (Kerberos) 197
6.4.2 El programa PGP 204
6.4.3 Seguridad con SNMP V2 207
6.5 Muros contra incendio (Fire Walls) 211

Conclusiones 219
Bibliografía 223

Introducción

El presente siglo se ha caracterizado por el desarrollo de los sistemas de comunicación, a nuestros abuelos les toco presenciar el gran desarrollo de las redes telegráficas y telefónicas, nuestros padres crecieron con los sistemas de radio y vieron nacer la televisión, mientras que a nosotros nos correspondió ser testigos de los grandes avances de las telecomunicaciones (los satélites, la fibra óptica, etc). Adicionalmente, nuestra generación también presenció el gran desarrollo de los sistemas de cómputo, y la fusión de estos con los sistemas de comunicación para crear las redes de computadoras. El hecho que las redes sean una fusión entre estas dos nuevas tendencias es lo que nos impulso a realizar el presente trabajo, consideramos a la red como un sistema integral, ya que conjuga aspectos de comunicaciones con sistemas electrónicos digitales.

Las redes son relativamente jóvenes; sin embargo, su complejidad ha generado diversos campos de interés. El presente trabajo se enfocará en la administración, enfatizándo los aspectos de seguridad; pero, debido a nuestra forma de ver las redes, el presente trabajo incluye algunos tópicos que por lo general no son considerados dentro de los libros que se encargan de análisis de la seguridad y administración de los sistemas de red. En nuestra opinión es necesario que el administrador de una red tenga por lo menos nociones de: conceptos fundamentales de las redes (topología, formas de control, estándares, etc), comunicaciones (medios de transmisión, relación entre el ancho de banda y velocidad de transmisión, etc) y conocimientos de software (sistemas operativos, programación, protocolos, servicios, etc).

Esperamos que esta tesis pueda ser una buena referencia para el estudio de las redes y sobre todo de los aspectos de interés (seguridad y administración), deseamos que los capítulos dedicados al análisis de aspectos generales de las redes sirvan como soporte a los que contienen nuestros puntos de interés propiamente dichos.

Capitulo I

Conceptos Generales

En los últimos tiempos el desarrollo tecnológico se ha centrado en la construcción de dispositivos que permitan establecer comunicación a largas distancias. Existe un viejo refrán que dice: Si la montaña no va a Mahoma, Mahoma va a la montaña; y ésta parece ser la tendencia que el desarrollo tecnológico ha seguido a lo largo de las dos últimas décadas. Los medios de transporte están tan desarrollados que el mundo se ha convertido en una aldea. Teóricamente el hombre puede viajar a cualquier parte, sin embargo las capacidades de los medios de transporte actuales parecen ser insuperables. Por otro lado, hemos tenido la suerte de observar el gran desarrollo de las comunicaciones, se han desarrollado sistemas mundiales de comunicación tales como la radio, la televisión o el teléfono, si ha esto le aunamos que el uso de las microondas, los enlaces vía satélite o los enlaces mediante fibra óptica han aumentado el poder de los medios antes citados, podemos pensar que en cierta medida el hombre ha podido superar la tiranía del espacio. La computadora es la otra invención reciente que ha modificado nuestra manera de ver el mundo. La construcción de sistemas digitales ha ayudado al desarrollo y perfeccionamiento de los sistemas de comunicación al igual que estas han influenciado el desarrollo de los sistemas de cómputo.

En los orígenes de la computación lo que se tenía eran sistemas centralizados, una máquina grande daba servicio a varios usuarios, con el paso del tiempo la tecnología fue avanzando de las máquinas grandes hacia las pequeñas lo que dio como resultado sistemas de cómputo personales. Gracias a los avances de los sistemas de comunicación se hizo posible la interconexión de los diferentes sistemas de computo.

Dicha interconexión ha permitido que los miembros de un equipo de trabajo ya no tengan que desempeñar sus labores en una misma sala, o en un mismo edificio; ahora pueden trabajar en ciudades o inclusive en países diferentes.

Por otro lado, la tendencia en los sistemas de cómputo parece revertirse y los usuarios de sistemas personales de cómputo necesitan de dispositivos y formas de cómputo que sólo se encuentran en las máquinas grandes. Como el costo de una computadora personal no es comparable al de una estación de trabajo o de una mainframe, es más económico establecer comunicación con una máquina grande ejecutar en ella el trabajo pesado y manejar los resultados en una computadora personal. Muchos investigadores de la Universidad Nacional Autónoma de México utilizan esta forma de cómputo. Mediante una conexión remota ejecutan complicadas operaciones de cálculo en una super-computadora (*CRAY*) y el análisis de los resultados obtenidos de la super-computadora se puede realizar en un sistema personal o en una estación de trabajo.

En la actualidad cualquier empresa que pretenda ser competitiva, sobre todo las encargadas de brindar servicios, debe contar con un buen sistema informático soportado en un buen sistema de comunicación. Además como dijera Eduardo Valtierra, director de Intel, "Si la computadora se encuentra aislada es un mueble más".

1.1 Concepto de RED

El estudio de las redes ha generado un gran interés en los últimos años. Se han publicado numerosos artículos, se han diseñado gran cantidad de estándar, sin embargo no podemos decir aún que el concepto de red está claramente definido. Los términos se utilizan en una variedad de formas, por lo tanto lo que algún autor puede considerar como una red para otro no lo es. Las definiciones dadas aquí son las que consideramos más generales y son las que se utilizarán a lo largo de este trabajo.

Al sistema formado por varios equipos de cómputo autónomos que se comunican entre sí para intercambiar información se le denomina RED de computadoras. Todas las redes de comunicación se forman por una colección de elementos de conmutación, elementos de cómputo y dispositivos periféricos, como se muestra en la figura 1.1.

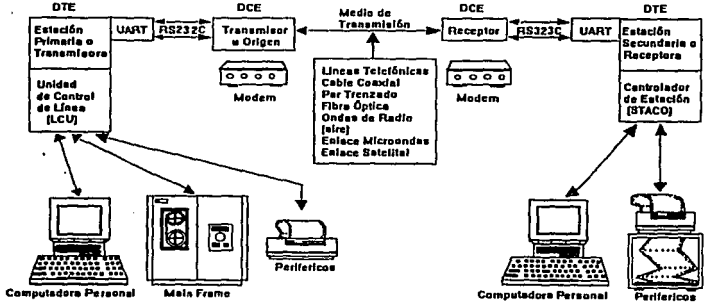


Figura 1.1 Sistema básico de una Red de computadoras

Es necesario hacer una distinción entre una red de computadoras y un sistema centralizado. Como ya mencionamos una red se forma por un conjunto de computadoras autónomas, por el contrario en un sistema centralizado una sola máquina es la que realiza el trabajo, esta máquina es capaz de trabajar por sí sola mientras que las máquinas que están conectadas a ella sólo funcionan como una extensión de la máquina central. En este tipo de sistemas a la computadora central, la que se encarga de desempañar todo el trabajo, es conocida comúnmente como MAINFRAME mientras que las máquinas que operan como satélites de la mainframe se les conoce como terminales tontas. Las mainframes están diseñadas principalmente para brindar una mayor capacidad de procesamiento, pero por lo general no son lo suficientemente flexibles para manejar demandas altamente interactivas.

También es necesario distinguir entre un sistema distribuido y una red. En una red el usuario debe realizar cualquier operación sobre una máquina remota de forma explícita. Por el contrario en un sistema distribuido las operaciones entre distintas máquinas son transparentes al usuario, el control de la información se ejecuta mediante el sistema operativo, lo que hace que el usuario de un sistema distribuido conciba al sistema como un sistema mono-procesador mientras que el usuario de una red está consciente que trabaja en un sistema formado por múltiples procesadores. Por lo tanto podemos decir que un sistema distribuido es un caso particular de una red de computadoras.

Para simplificar el estudio, la construcción y el diseño de las redes se hace una división entre los aspectos puros de comunicaciones y los aspectos de las aplicaciones. La parte encargada de las aplicaciones se conoce como HOST y la parte que se encarga de las comunicaciones se le conoce como SUB_RED abreviatura de sub_red de comunicaciones.

Host.- Colección de máquinas que intentan ejecutar los comandos y programas del usuario. En algunas ocasiones también se usa el término de sistema terminal o sistema final como un equivalente al de host. El host debe ser capaz de determinar su posible acceso a la red. El acceso a la red se efectúa mediante un procesador encargado específicamente de esta función y una unidad de medio de conexión (Medium Attachment Unit MAU). Como por lo general los elementos conectados dentro de una red no provienen de un solo fabricante no tienen un protocolo común de comunicación. Por lo tanto, una de las principales funciones de los controladores de acceso es la de transformar el protocolo de transmisión de los dispositivos en el protocolo utilizado por la red. Otras funciones del controlador de acceso a la red son: enviar los datos de un equipo hacia la red hasta que la red este disponible, revisar cada uno de los mensajes dentro de la red para ver si está destinado al dispositivo al que está conectado, leer los mensajes dentro de su buffer y vigilar las señales de entrada para asegurarse de que no existan errores.

Sub_red de comunicaciones.- La sub_red de comunicaciones se encarga de transportar los mensajes de una máquina a otra, de la misma forma en que el sistema telefónico envía los mensajes del emisor al receptor. La sub_red de comunicaciones a su vez se divide en líneas de transmisión (canales) y los elementos de conmutación (nodos). Las líneas de transmisión (a las que también se les conoce como circuitos o troncales), se encargan de mover bits entre las máquinas.

Los elementos de conmutación son máquinas especializadas que se utilizan para conectar dos o más líneas de transmisión. Su función es la de establecer un patrón de comunicación entre diferentes equipos y dirigir la información dentro de la red.

A los elementos de conmutación también se les conoce como IMP (procesadores de intercambio de mensajes). Todo el tráfico que va o viene del host pasa a través de su IMP.

Desde el punto de vista del usuario un mensaje en la red es una unidad de comunicación. Por ejemplo en el correo electrónico un mensaje puede ser un documento enviado de un usuario a otro. Para poder transmitir un mensaje a través de la red este debe ser representado como una cadena de símbolos , esto es unos y ceros. Estos símbolos binarios es lo que conocemos comúnmente como bits (BInary digiTs). Para transmitir estos mensajes dentro de la red se necesitan bits adicionales para asegurar una comunicación confiable, corregir la ruta de los mensajes y prevenir la congestión de la red. Adicionalmente, no es común transmitir los mensajes como una unidad completa ya que esto generaría retraso en el funcionamiento de la red, problemas para el control del buffer así como problemas para el control del congestionamiento de la red. Por lo tanto, los mensajes se rompen en pequeñas cadenas de símbolos llamadas paquetes (pakets). Los paquetes se envían a través de la red como unidades individuales y son reconstruidos para formar el mensaje en la unidad receptora (host).

1.2 Objetivos de las redes

Todas las redes se construyen esencialmente por las mismas razones:

- La posibilidad de compartir recursos.
- Mejorar la confiabilidad de los sistemas.
- Reducir costos.
- Como medio de comunicación.

1.2.1 Compartir recursos

Uno de los objetivos principales de una red es hacer que todos los programas, datos y equipo estén disponibles para cualquier usuario de la red que así lo solicite, sin importar la localización física del recurso y del usuario. En otras palabras, el hecho de que el usuario se encuentre a 1000 km de distancia de los datos, no debe evitar que éste los pueda utilizar como si fueran originados localmente.

1.2.2 Mejorar la confiabilidad de los sistemas

Este es otro de los objetivos de las redes. Cuando sólo existe una computadora, como en el caso de los sistemas centralizados, una falla en algún componente de la misma hace que ésta salga de operación y por lo tanto todos los usuarios se ven afectados. Una red con muchos equipos, mejora la confiabilidad ya que si un equipo falla, en redes bien diseñadas, los demás equipos pueden continuar trabajando. Los archivos importantes pueden ser almacenados en varios equipos; de tal manera que si un equipo falla los datos puedan ser obtenidos de otra fuente. Los usuarios tienen la posibilidad de utilizar varios sistemas.

1.2.3 Reducir costos

Una mainframe típica dentro de un ambiente de oficina es entre diez y cien veces más eficaz que una computadora personal, pero cuesta entre cien y mil veces más que una PC. De hecho los microprocesadores tiene un mejor radio de costo/operación que la mayoría de las mainframes. En un ambiente de oficina típico, donde los usuarios usan aplicaciones tales como procesadores de palabras u hojas de cálculo, una red con cien computadoras personales puede ser más efectiva y económica que una mainframe con cien terminales, y su respuesta en el tiempo será indudablemente mejor.

1.2.4 Como medio de comunicación

Con el empleo de una red es relativamente fácil para dos o más personas, que viven en lugares separados, escribir un informe juntos. Cuando un autor hace un cambio en un documento que se mantiene en línea, los otros pueden ver el cambio de inmediato, en lugar de esperar varios días para recibirlo por carta. Esta rapidez hace que la cooperación entre grupos de individuos que se encuentran alejados, y que anteriormente había sido imposible de establecer, pueda realizarse ahora.

1.3 Clasificación de las redes

Las redes pueden ser clasificadas en diferentes formas. La forma más común de clasificarlas es en base a su aplicación principal, a su alcance y velocidad.

Redes de área local (Local Area Networks LAN).- Este tipo de red es utilizada generalmente para interconectar diferentes equipos de cómputo dentro de un área pequeña, tal como una oficina un edificio o un pequeño grupo de edificios (Campus). Las redes de área local típicamente operan a velocidades de 20 millones de bits por segundo (Mbps), conectando cientos de dispositivos que se encuentran en una distancia de entre 5 y 10 kilómetros.

Existe un tipo especial de redes de área local de alta velocidad conocidas como HSLN (High Speed Local Networks). Las HSLN se utilizan para conectar un pequeño número de computadoras y periféricos de alta velocidad. Su aplicación está típicamente limitada a distancias de 1 km como máximo, y operan a velocidades entre 50 y 100 Mbs. La red de la Facultad de Estudios Superiores Cuautitlán (Campo 4) conocida como Cuautitlán 2 es un ejemplo de una red de área local.

Redes de área metropolitana (Metropolitan Area Networks MAN) .- Redes que engloban numerosas poblaciones dentro de un área mediana tal como una ciudad o estado. Las redes de área metropolitana representan el primer esfuerzo para diseñar otro tipo de redes más ambicioso, las redes de área amplia. A grosso modo podemos decir que una red de área metropolitana es la interconexión de redes de área local. Las primeras redes de este tipo operaban a velocidades de 45 a 150 Mbps y tenían una cobertura de aproximadamente 100 km. Como ejemplo de red de área metropolitana tenemos la red de la Universidad Nacional Autónoma de México, conocida como RedUNAM. Esta red une las diferentes redes de área local pertenecientes a los institutos y facultades que conforman la Universidad Nacional Autónoma de México. En el capitulo VI se analizará en más detalle la RedUNAM dando mayor énfasis a la red Cuautitlán 2. En términos generales la RedUNAM tiene 3 nodos principales encargados de formar la columna vertebral (backbone) de la misma, los tres nodos principales son: La Dirección General de Servicios y Cómputo Académico (DGSCA) el instituto de Astronomía (ASTROS) y el instituto de Investigaciones en Matemáticas Aplicadas y Sistemas (IIMAS). El objetivo de la RedUNAM es optimar los recusrsos de cómputo existentes en las diferentes Facultades e Institutos, así como facilitar la conexión de los mismos con otras instituciones internacionales.

Redes de área amplia (Wide Area Networks WAN).- Las redes de este tipo extienden su cobertura a aplicaciones a nivel nacional e internacional. La velocidad de las redes de área amplia varía desde las lentas de 120 bps hasta las más moderadas de 45 Mbps.

Ejemplos de redes de área amplia de comunicación pueden ser la red telefónica o bien la red de televisión y radio. Estas redes no son redes de computadoras, pero las soluciones implementadas para su desarrollo son útiles para el desarrollo de las redes informáticas.

La red de área amplia más importante en la actualidad es Internet. Esta red esta formada aproximadamente con 39,000 redes registradas en 107 países, con 2,500,000 nodos y al menos 3,500,000 usuarios activos. La red Internet al tener una base tan grande a nivel mundial tiene objetivos muy claros y específicos como facilitar la posibilidad de compartir recursos entre las organizaciones participantes, como son las agencias de gobierno, instituciones educativas y corporaciones privadas; así como promover el interés y participación de investigadores y proveerles de un ambiente de prueba para nuevos desarrollos en redes de comunicación.

1.4 Tipos de redes

Las redes también pueden ser clasificadas de acuerdo al tipo de canales de comunicación que utiliza la sub_red de comunicaciones (sub_red) que la forma. En base a este parámetro existen dos tipos de redes:

- Redes punto a punto o de conmutación (point to point).
- Redes de transmisión por canal o multipunto (broadcast).

1.4.1 Red punto a punto

En una red punto a punto los mensajes enviados de una máquina (host) a otra se pasan de forma serial a cada nodo de acuerdo al patrón entre las máquinas. La figura 1.2 muestra una red de conmutación con varias rutas posibles entre dos máquinas. Existen dos tipos de redes de conmutación: las de circuitos de conmutación y las de almacenar y seguir. Estos tipos de redes multipunto serán discutidos más adelante.

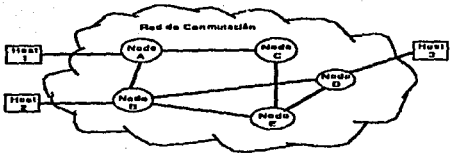


Figura 1.2 Hosts (máquinas) y nodos en una Red de Conmutación

1.4.2 Red de transmisión por canal

En las redes de transmisión por canal, todos las máquinas están conectadas a un medio común, como se muestra en la figura 1.3. En este tipo de redes la información no está dirigida lo que trae como consecuencia la necesidad de un medio de control que asegure que solamente una unidad está transmitiendo en un momento determinado. Las redes de transmisión por canal deben utilizar alguna clase de sistema de multiplexado para permitir que todos los posibles dispositivos transmisores compartan la sub_red de comunicación.

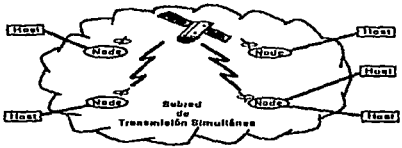


Figura 1.3 Hosts (máquinas) y nodos en una Subred de Transmisión Simultánea.

1.5 Multiplexado

El multiplexado en una red permite que un mismo recurso sea compartido por varios usuarios. Los multiplexores en una red telefónica, por ejemplo, permiten mantener muchas conversaciones a través de una misma línea de comunicación.

En las redes de datos y las de telecomunicaciones se emplean dos tipos de esquemas de multiplexado:

- División de frecuencia.
- División de tiempo.

1.5.1 Multiplexado por división de frecuencia

El esquema de multiplexado por división de frecuencia (Frequency Division Multiplexing FDM) es un método que permite que un mismo canal de comunicación sea compartido por múltiples usuarios mediante la asignación de porciones específicas del espectro de frecuencia de transmisión del canal a cada uno de los usuarios, como se muestra en la figura 1.4.

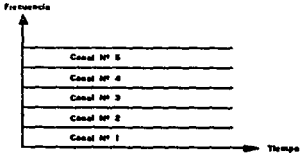


Figura 1.4 Multiplexado por División de Frecuencia

El esquema FDM es el que se utiliza en la transmisión de las señales de televisión. Cada una de las estaciones de televisión requiere de seis millones de hertz o seis megahertz (Mhz) y todas las estaciones comparten el espectro de frecuencias disponibles en su medio de transmisión, el aire. El sintonizador del televisor actúa como un demultiplexor para sintonizar solamente la banda utilizada por el canal especificado por el usuario. Este mismo principio es empleado por otros sistemas analógicos de comunicación tales como la radio o el sistema de televisión por cable.

La red analógica de teléfono también usa FDM para permitir múltiples conversaciones a través de los mismos medios. En la red normal telefónica, una sola voz ocupa toda la amplitud del canal de 300 a 3400 Hz. En una línea compartida de una red telefónica, cada conversación se coloca en una banda diferente de 3.1 kHz. Como el tamaño del canal (el ancho de banda) es constante, la integridad de la información a través de la red se mantiene a pesar de que la frecuencia de transmisión de los datos haya sido alterada.

1.5.2 Multiplexado por división de tiempo

Las señales digitales en un medio común por lo general son multiplexadas usando multiplexado en el tiempo (Time División Multiplexing TDM). Mientras que el esquema FDM asigna parte del espectro de frecuencia a cada usuario por todo el tiempo que éste lo requiera, el esquema TDM asigna al usuario todo el espectro de frecuencias durante pequeños espacios de tiempo.

La figura 1.5 muestra como 5 usuarios tienen acceso a la red de comunicación en diferentes períodos de tiempo sin dar preferencia a ninguno de ellos. En algunas ocasiones el multiplexado de la figura se conoce como TDM síncrono porque si conocemos el período específico en el que aparece el dato en la red implícitamente sabemos que canal está transmitiendo. El TDM síncrono tiende a

desperdiciar el ancho de banda, tiene asignado un cierto número de intervalos, donde cada uno de estos intervalos representa un usuario o canal e independientemente de si el canal está ocupado o no, el sistema le respetará el tiempo que tiene asignado para uso del medio. De esta manera durante los intervalos asignados a canales no utilizados el medio estará inactivo, lo que reduce la eficiencia del medio de transmisión cuando no están ocupados todos los canales.

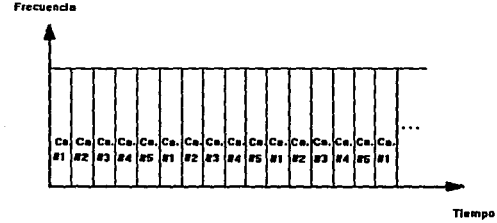


Figura 1.5 TDM Síncrono.

Otro tipo de esquema TDM es el TDM asíncrono, o esquema de multiplexado estático. En el multiplexado estático se le asigna un intervalo de tiempo tan sólo a los canales activos, como se muestra en la figura 1.6. De esta manera los medios de comunicación estarán disponibles tan sólo para canales activos, el medio de comunicación permanecerá inactivo sólo cuando todos los canales estén sin utilizar.

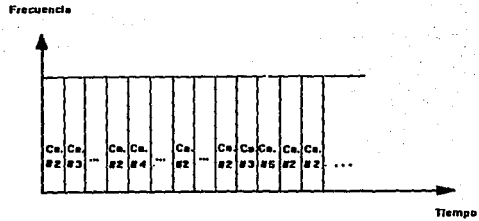


Figura 1.6 TDM Estadístico.

1.6 Tipos de redes de conmutación

Para comprender de una manera más completa el funcionamiento de las redes, es necesario entender los dos tipos de conmutación que se utilizan para el manejo de la información. Los tipos de conmutación que existen son el de circuitos de conmutación (por lo general utilizados para manejar voz) y el de paquetes de conmutación (comúnmente utilizados para el manejo de datos). Ambas técnicas de conmutación se utilizan en la actualidad y sus aplicaciones serán soportadas de una forma u otra por las redes, sobre todo en las de área metropolitana, que son las que deben manejar tanto voz como datos.

1.6.1 Circuitos de conmutación

Los circuitos de conmutación es la técnica de conmutación más popular; la red telefónica es un ejemplo de este tipo de red. En una red de circuitos de conmutación, la vía de comunicación entre dos

usuarios estará disponible durante todo el tiempo que sea necesario y no será compartida con otros usuarios. Aunque la red telefónica utilice FDM para hacer que varios usuarios compartan un mismo canal de comunicación su forma de trabajo puede considerarse como de circuitos de conmutación ya que sólo existe un usuario asignado a cada canal.

Una conexión mediante circuitos de conmutación se establece hasta que dos usuarios generan una vía de comunicación entre ellos. La ruta se establece hasta que uno de los usuarios inicia el procedimiento para establecer la llamada mediante el cual informa al sistema la dirección de red del usuario al cual desea llamar (por ejemplo hasta que un usuario marca el número telefónico de otro). Cabe aclarar que la vía de comunicación no será necesariamente la misma cada vez que estos dos usuarios establezcan comunicación entre ellos.

La conexión a través de la red entre los usuarios durará el mismo tiempo que dure la llamada. Durante el tiempo que esta dure, el circuito será equivalente a un par de cables conectando los sistemas de comunicación de ambos usuarios. La conexión física del circuito está dedicada solamente a esta llamada y no es compartida con otros usuarios.

Las conexiones mediante circuitos de conmutación son funcionales para el manejo de tráfico de voz. La vía de comunicación es sensitiva al silencio, es decir que no puede agregar o remover los periodos de silencio del mensaje. El tiempo perdido en el establecimiento de la llamada es compensado por su facilidad de mantener la comunicación durante grandes periodos de tiempo.

Por razones similares, este tipo de conexión no son eficientes para transmitir datos. Los datos no se transmiten de forma continua sino mediante ráfagas, por lo tanto un procedimiento muy largo para establecer la llamada significa pérdida de tiempo. Como la red telefónica está optimada para transmitir voz, todos los canales son de banda estrecha; esto significa que las llamadas para la

transmisión de datos duraran más tiempo. Además, dedicar un canal para la transmisión de ráfagas significa tener el canal inactivo la mayor parte del tiempo.

1.6.2 Conmutación de paquetes

Para este tipo de conmutación no existe una línea dedicada a la conexión de dos equipos, los usuarios ponen sus mensajes en la red para que esta se encargue de llevarlos a su destino. De esta forma la comunicación entre los usuarios se efectúa de forma lógica en lugar de forma física. Como los canales físicos no están dedicados a una conexión punto a punto, pueden ser compartidos por múltiples usuarios. De esta manera, la conmutación de paquetes optima el uso de los recursos de la red asegurándose que los canales físicos nunca estarán inactivos, salvo los períodos en los que no exista tráfico en la red. Tradicionalmente, la conmutación de paquetes sólo es aplicable a tráfico lento, sin embargo tecnologías de manejo de paquetes más rápidas, como las establecidas en la capa de marco del estándar I.122 de la CCITT están cambiando el tipo de aplicación de la conmutación de paquetes.

Como ya mencionamos para la transmisión de mensajes la red los divide en paquetes. El paquete tiene un tamaño máximo que usualmente es de 128 o 256 octetos.

Una conexión de conmutación de paquetes establece una vía lógica entre dos máquinas pero sin dedicar un medio físico exclusivamente a esta conexión. De esta manera, varias conexiones de conmutación de paquetes pueden compartir un mismo medio de transmisión, optimando el uso de los recursos de la red. Cuando el paquete es recibido por un nodo se coloca en un buffer y después es enviado al siguiente nodo en el patrón lógico de comunicación. El control del tráfico en la red se hace mediante técnicas de multiplexado por división en el tiempo estáticas.

Los paquetes enviados por un nodo a través de la red son pasados de nodo a nodo hasta que encuentran su destino. El nodo que contenga el paquete debe conservar el paquete hasta que pueda enviarlo al siguiente nodo; por está razón, la conmutación de paquetes se conoce como estrategia de almacenar y seguir.

El principal problema con las redes de conmutación que siguen la estrategia de guardar y seguir es que pueden hacer que algunas transmisiones se retrasen. Por ejemplo si varios paquetes están listos para ser transmitidos por el canal de comunicación al mismo tiempo, el nodo transmitirá uno de ellos y mantendrá el resto en el buffer. Al menos que el tiempo de almacenamiento sea excesivamente alto, el retardo no será un problema para la mayoría de las aplicaciones. Las aplicaciones que no puedan incluir períodos de silencio ni removerlos no pueden usar la estrategia de almacenar y seguir excepto bajo las condiciones más ideales.

Cuando dos máquinas (hosts) se comunican mediante una red de conmutación de paquetes (Packet Switching Network PSN), típicamente lo hacen mediante un circuito virtual entre ellos definido por la conexión lógica entre las máquinas. A pesar de que todos los paquetes asociados con el circuito virtual probablemente seguirán la misma ruta a través de la red, ningún usuario posee una línea física. Por ejemplo, puede observarse en la figura 1.2 que un circuito virtual entre las máquinas 1 y 3 y uno entre las máquinas 2 y 3 pude compartir la ruta entre los nodos B y D.

1.6.3 Circuitos virtuales y datagramas

En el mundo de las comunicaciones se tienen dos tipos de servicios, los circuitos virtuales y los datagramas. Establecer un circuito virtual es similar al procedimiento de efectuar una llamada telefónica. Alguno de los usuarios debe iniciar un procedimiento para entablar comunicación con otro.

Durante este proceso, la máquina que inicia la comunicación debe proporcionar la dirección de la máquina con la que desea comunicarse y la red debe establecer la ruta para los paquetes. Como el establecimiento de la ruta se hace sólo una vez, todos los paquetes seguirán el mismo patrón; de esta forma la red pude garantizar que todos los paquetes llegarán a su destino y que lo harán en forma secuencial. Si la red se congestiona o si se detectan errores, los nodos pueden manejar la situación, son capaces de identificar a los circuitos que están ocasionando el problema. El servicio de circuito virtual se conoce también como patrón de comunicación enfocado a conexión (Conection Oriented).

Un datagrama puede definirse como un paquete de longitud finita con información suficiente para ser enviado en forma independiente de la fuente al destino, sin recurrir a transmisiones anteriores. Una conexión mediante datagramas es análoga a enviar una carta. No se requiere un procedimiento para establecer la comunicación entre dos máquinas. Todos los paquetes, sin embargo, deben contener la dirección de su destino y se envían de forma independiente. Por lo tanto, no existe garantía de que todos los paquetes llegarán a su destino ni de que llegarán de forma ordenada; lo que ocasiona, que en redes con detección de errores, algunos paquetes puedan ser duplicados. La congestión de la red y los paquetes con errores son manejados por los nodos descartando paquetes; los nodos usualmente no tienen forma de saber de donde viene los datagramas; por lo tanto, manejan los errores descartando los paquetes que estén sobrecargando el sistema, las máquinas receptora y emisora se encargan del manejo de los errores en los mensajes. Por razones obvias el sistema de datagramas se considera un sistema no orientado a conexión (connectionless)[1]. Normalmente cuando dos mensajes se envían al mismo destino, el primero que se envíe será el primero en llegar. Es posible, sin embargo, que el primero que

[1] Quiza una traducción mejor para el término "connectionless" es la de no conformado(a). En algunos textos lo traducen como "inseguro", esto se refiere a que no se tiene seguridad que los paquetes llegarán a su destino, no en el sentido de privacidad de la información.

se envie sufra un retraso y llegue antes el que se envío en segundo lugar; por el contrario, en un servicio orientado a conexión es imposible que suceda esto. Cada servicio se caracteriza por su calidad; algunos de ellos son fiables en la medida que nunca pierden la información que transportan. Por lo general, un servicio fiable se realiza haciendo que el receptor notifique haber recibido cada mensaje, para que el transmisor esté seguro de que su mensaje llegó a destino. El proceso de notificación introduce un exceso de tráfico y retardos, que a menudo son convenientes, pero también son algunas veces indeseables.

1.7 Topologías

La forma en que están conectados los nodos es lo que se conoce como topología de una red. Las redes de área local están caracterizadas de acuerdo a su topología. Por lo tanto, para determinar la topología que mejor se adecue a las necesidades de un sistema en particular hay que tener en cuenta diferentes aspectos, tales como: localización, número y distribución de las computadoras, ancho de banda requerido para soportar el tráfico que demanden los usuarios, en caso de que existan una o más maquinas grandes, que topologías son las que soportan para una mejor integración y finalmente pero no menos importante el costo tanto en hardware y software así como el cableado.

1.7.1 Control

La principal implicación de la topología de una red es la forma como las estaciones participan en el proceso de obtener permisos para utilizar el medio de transmisión, ésto se conoce como control de la red. El control puede ser distribuido o centralizado.

Si dos dispositivos, conectados de forma punto a punto, comparten un sólo canal de comunicación, el control es relativamente simple, cuando una estación transmite la otra recibe, si la estación transmisora requiere una respuesta de la receptora los papeles se invierten y la transmisora pasa a ser receptora y la receptora se vuelve transmisora.

En un ambiente de transmisión por canal (Broadcast), los procesos de control son más complicados. Si en un sistema de este tipo dos estaciones o más intentan transmitir al mismo tiempo, toda la información resultará dañada. El control establece un mecanismo mediante el cual se decide que estación puede transmitir en un momento determinado.

Como ya mencionamos las redes pueden tener dos estrategias de control, centralizado o distribuido. La estrategia de control centralizado asigna una estación como primaria, o controladora, y designa a las demás como secundarias. En este tipo desbalanceado de configuración la primaria puede transmitir información a cualquier secundaria en cualquier momento; pero la estación secundaria tan sólo puede transmitir a la primaria y solamente puede hacerlo cuando la primaria le ha dado explícitamente permiso de hacerlo. Las redes de área local por lo general no utilizan este tipo de control.

El control distribuido coloca a todas las estaciones como iguales, pueden transmitir toda la información que quieran y a la estación que quieran. El esquema de control distribuido necesita de un cierto grupo de reglas común para todas las estaciones, este grupo de reglas tiene como objetivo establecer una distribución justa de acceso a la red y asegurarse que sólo una estación transmite a la vez. Como se trata de una configuración balanceada todas las estaciones deben seguir el mismo grupo de reglas (ya que no existe ninguna estación que controle la comunicación). Casi todas las redes de área local utilizan esta estrategia de control. Las reglas que gobiernan el acceso a la red están definidas

en varios estándares de control de acceso a los medios (Medium Access Control MAC). Los cuales serán discutidos en el capítulo III.

1.7.2 Modelos de topologías

Las topologías pueden dividirse en dos grandes categorías, las regulares y las irregulares. Las redes de área local; por lo general, tienen topologías geométricas mientras que las redes de área metropolitana y las de área amplia usan topologías irregulares.

Las tres topologías regulares más comunes son: la de bus, la de estrella y la de anillo. Las cuales se explican a continuación.

	Irregular	
Modelos de		Bus
Topología	Regular	Estrella
		Anillo

Topología de bus .- En este tipo de topología todos los nodos de la red están conectados a un medio de transmisión común. Como resultado de esto, sólo dos usuarios pueden comunicarse en un momento dado. Cada nodo de la red tiene una dirección única, la cual se utiliza cuando se realiza la transmisión. Cuando se envía un paquete, este se propaga a través del medio y es recibido por todas las estaciones. Para recibir mensajes, cada estación continuamente monitorea el medio, copiando los mensajes que estén dirigidos a ella y dejando pasar los que no lo estén. Su operación es similar a la de un volmetro, este mide el voltaje de una línea sin interrumpir el flujo de electrones.

Las estaciones conectadas con la topología de bus son análogas a los aparatos eléctricos utilizados en los hogares, los cuales se conectan a la línea de corriente alterna. Todos los dispositivos extraen energía de la línea sin importar que estén conectados a diferentes segmentos de la misma, operando de forma independiente. Un diagrama de la topología de bus se puede observar en la figura 1.7.

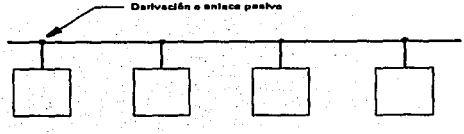


Figura 1.7 Topología en Bus.

La conexión de bus genera una red de transmisión por canal simultánea, esto significa que todas las estaciones reciben el mensaje transmitido esencialmente al mismo tiempo (ignorando el retardo de propagación de la señal en el medio).

Existen, básicamente, dos tipos de topologías de bus. En la topología de bus banda base (baseband), el transmisor aplica directamente las señales al bus sin modificarlas o modularlas. Los bits en un bus banda-base son transmitidos unidireccionalmente y no pueden ser modificados por los receptores. El otro tipo de bus es el conocido como bus de banda ancha (broadband), usa señales moduladas en una frecuencia determinada por el transmisor; estas señales se propagan hacia un dispositivo que se encarga de retransmitir la señal hacia las estaciones receptoras.

La topología de bus es probablemente la topología de red más antigua; la primera red de área local descrita en las publicaciones científicas fue la red Ethernet, la cual uso la tecnología de bus, de banda ancha.

Como cualquier arquitectura de red la topología de bus tiene ventajas y desventajas. Sus ventajas más significativas son que el tirado de cable es más pequeño que en otras topologías, está arquitectura idealmente se utiliza para trabajar con protocolos de contención de alta velocidad, se presta para cambiar la configuración del sistema (añadir y remover usuarios) y el medio de transmisión es altamente confiable (la falla de uno de los sistemas no afecta la operación de los demás). Además el hardware para bus de cable coaxial es fácil de conseguir ya que ha sido utilizado por algunos sistemas de televisión así como por sistemas telefónicos y de radio.

Las limitaciones de la topología de bus incluyen las restricciones en las distancias de transmisión, la dificultad que presentan para establecer prioridades a alguna máquina y problemas con el balanceo de la señal. Los problemas de balanceo de la señal se generan cuando se utiliza la topología de bus en una línea multipuerto. Si dos usuarios distantes se desean comunicar, la señal a ser transmitida debe ser lo suficientemente fuerte como para tener una relación señal/ruido baja en el receptor. Sin embargo, si el mismo transmisor desea comunicarse con una estación cercana, la señal de transmisión no debe ser tan fuerte como para sobrecargar al receptor. Si existen n estaciones en la red y se realiza el balanceo de la señal para todas las posibles conmutaciones entre las estaciones tomando dos a la vez, entonces se deben realizar $n(n - 1)$ balances. Para un gran número de estaciones el balanceo puede resultar un problema tedioso o difícil de manejar.

La topología de bus es muy usada en la construcción de redes de área local. Las aplicaciones de las redes de área local con topología de bus son: automatización de oficinas, control de procesos

industriales, acceso a bases de datos, transferencia de información, conexiones de múltiples equipos de cómputo, así como la transmisión integral de voz, datos y video, etc.

Topología de estrella .- En la arquitectura de estrella todos los nodos se juntan en un solo punto conocido como nodo central o eje. El nodo central puede ser un dispositivo activo o pasivo. Tal como se muestra en la figura 1.8.

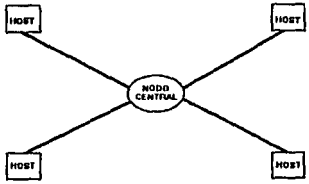


Figura 1.8 Topología en Estrella.

Los ejes activos se utilizan cuando el control de la red debe ser manejado por un nodo en particular. En este caso el nodo central se encarga de establecer todas las rutas de los mensajes dentro de la red, ya sea del nodo central a los nodos exteriores, entre los nodos exteriores, o de todos los nodos hacia puntos remotos. Los nodos exteriores y el nodo central están unidos mediante enlaces punto a punto. Las redes estrella con un nodo central activo son muy útiles cuando la mayor parte de la comunicación es entre el nodo central y los nodos exteriores. Si el tráfico entre los nodos exteriores es continuo, entonces el nodo central se encontrará muy cargado.

En una estrella con un eje pasivo, se utiliza un divisor de señal como eje de la estrella para dividir la señal de entrada entre todas las estaciones. Las pérdidas en el divisor pueden sumar hasta 1/N

donde N es el número de ramas en el divisor. De esta manera, si existe un gran número de terminales conectadas será necesario tener amplificadores para elevar el nivel de la señal, de forma que pueda ser transmitida eficientemente.

Los mejores ejemplos de una red en estrella son los conmutadores de teléfonos controlados por computadora (Private Branch eXchanges PBX) y los conmutadores de datos (data switches). Los PBX conectan los teléfonos directamente al conmutador (switch). Cuando un usuario desea llamar a otro, la dirección del destinatario (el número telefónico o el número de la extensión) se envía al conmutador, el cual provee un patrón de comunicación entre dos teléfonos. A pesar que no se puede establecer comunicación sin la ayuda de los PBX, estos no controlan la comunicación punto a punto. Los PBX no se consideran redes de área local ni son comúnmente usados para el manejo de datos, sin embargo los conmutadores de datos trabajan de forma similar para interconectar servidores, terminales y computadoras personales.

La topología de estrella tiene la gran ventaja de hacer la administración y manejo de la red relativamente sencillos. Sus desventajas incluyen el alto costo del nodo central y de su dependencia respecto a éste, si el nodo central queda fuera de servicio toda la red queda inutilizada.

Topología de anillo .- En la topología de anillo los nodos se conectan en forma consecutiva mediante enlaces punto a punto arreglados de tal forma que formen un patrón cerrado tal como se muestra en la figura 1.9. La información en forma de paquetes se transmite de nodo a nodo a través del anillo. Cada nodo es un dispositivo activo que tiene la habilidad de reconocer su dirección dentro de los paquetes para de esta manera poder recibir mensajes. Los nodos no sólo sirven como puntos de unión sino que son a la vez repetidores de las señales destinadas a otros nodos dentro de la red.

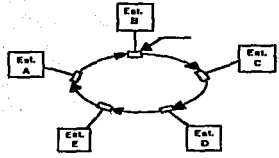


Figura 1.9 Topología en Anillo.

Los nodos en una red de anillo pueden operar de tres formas: modo de receptor, modo de transmisor y modo de paso. En la figura 1.10 se muestra el diagrama de cada uno de estos modos de operación. Cuando se encuentra en modo de receptor el nodo busca en todos los mensajes la señal que le permita saber que el mensaje está destinado a el. La cadena de bits recibidos se envían a través de la salida con un retardo de un solo bit. Cuando algún nodo detecta que el mensaje está dedicado a el, cada uno de los bits se envía a la estación conectada al nodo mientras que simultáneamente los bits son retransmitidos a la siguiente estación.

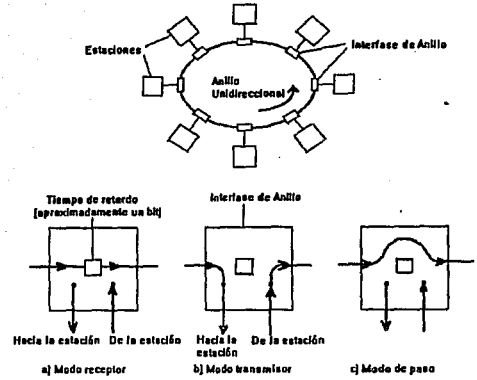


Figura 1.10 Topología en Anillo y sus tres posibles modos de operación.

Esto genera la pregunta, de cuando o donde deben ser removidos los mensajes del anillo. En la arquitectura de bus las señales que viajan a través de la línea son absorbidas por los terminadores de la línea una vez que llegan al punto final del bus. Sin embargo con una topología de anillo los mensajes podrían permanecer en circulación indefinidamente si no son absorbidos por alguna de las estaciones. Esto puede hacerse en la estación receptora o bien puede hacerse en la estación transmisora una vez que el mensaje ha dado una vuelta completa a la red. El que la estación transmisora sea la que remueva la señal de la red resulta ser más ventajoso ya que de esta manera puede garantizarse que el mensaje fue recibido y además permite enviar un mismo mensaje a más de una estación.

En el modo transmisor, el nodo rompe la conexión entre la entrada y la salida, para añadir su información al mensaje que viaja a través de la red. Existen varios métodos para cambiar del modo de

receptor al de transmisor, implementados mediante diferentes estándares de control de acceso a los medios (Medium Access Control MAC) que además se aseguran que sólo una estación transmita a la vez.

Para configurar una red de anillo se deben tener en cuenta varias cosas. Primera, todos los nodos en el anillo deben estar perfectamente conectados. Cuando se añade un nodo al anillo, se deben poner cables de transmisión entre este nodo y los dos nodos adyacentes. Debido a esto, es difícil planear futuros nodos en una red de anillo. Además la falla en alguno de los nodos, en alguno de los cables o la instalación de un nuevo nodo (durante el tiempo que tarde la instalación) harán que la red quede fuera de servicio. Existe una gran variedad de soluciones a estos problemas, sin embargo la construcción de estas soluciones por lo general aumenta la complejidad de las interfaces de los nodos y por lo tanto se incrementa su costo.

Topología irregular .- Las redes de topología irregular, también conocidas como redes híbridas deben su nombre a que no tienen una topología definida. En lugar de tener nodos conectados de forma punto a punto los enlaces se hacen de forma arbitraria y pueden variar de un modelo a otro. Las conexiones se determinan usualmente en base al costo de la red. Por ejemplo, cuando los costos de una línea de transmisión son muy altos y sólo unos cuantos nodos deben comunicarse entre ellos, se puede obtener una mayor eficiencia estableciendo sólo las conexiones entre los nodos que sean realmente necesarias.

A pesar de que la topología irregular brinda una mayor flexibilidad para la configuración de la red, los problemas para dirigir la información a través de la red se vuelven muy complicados. Los nodos encargados de establecer la ruta de la información dentro de la red deben ejecutar muchas funciones relacionadas con la red.

1.8 Control de Acceso al Medio (MAC)

Como ya hemos mencionado, las redes de área local por lo general son redes de transmisión por canal que conectan dispositivos que tienen los mismos derechos de acceso a la red. Esto genera dos requisitos en el protocolo que controla la red. Primero, debe haber tan sólo una estación transmitiendo a la vez, ya que transmisiones simultáneas podrían ocasionar corrupción en la información. Segundo, todas las estaciones deben seguir las mismas reglas de acceso ya que no se cuenta con ninguna estación maestra.

A los diferentes métodos para el control de acceso a la red se les conoce como esquemas MAC. A pesar de que existen diferentes tipos de MAC, en esencia son variantes de dos estrategias, conocidas como: "contención" y "probar y seguir" (Token passing).

El MAC define las reglas para acceder a la linea de la red y se encarga del montaje de los paquetes de datos. Se añade información de cabecera y cola a un paquete de datos para identificar el comienzo y el final de un mensaje, la información sobre encadenamiento y los controles de detección de errores.

1.8.1 Contención

El esquema de contención puede ser comparado con un grupo de personas discutiendo en una mesa, pero sin contar con un moderador. Cuando alguien desea hablar, lo primero que debe hacer es asegurarse que nadie más esté hablando; si alguien se encuentra hablando, entonces debe esperar a que haya silencio para poder tomar la palabra. Si dos personas empiezan a hablar al mismo tiempo ocurre una colisión. Las colisiones se resuelven de dos formas; o bien las dos personas se callan y esperan a

que el otro tome la palabra, (forma cortés) o bien ambos continúan hablando cada vez más fuerte hasta que uno de los dos se da por vencido (forma ruda).

El esquema de contención comúnmente usado en las redes de área local es muy similar a la situación descrita arriba (semejante a la forma cortés) y se conoce como acceso múltiple tras verificación de ausencia de portadora, con detección de colisiones (Carrier Sense Multiple Access with Collision Detection CSMA/CD). De las estrategias de control, ésta es una de las más antiguas, muchos de los esquemas de contención actuales pueden considerarse como variantes del CSMA/CD.

La estrategia CSMA/CD sólo es apropiada para redes de bus lógico. Cuando una estación está lista para transmitir, primero sensa el medio de transmisión, si detecta una señal sobre la línea, continuará monitoreando el canal. Una vez que detecta silencio en la red, entonces envía su mensaje. Las estaciones probarán continuamente el canal, si se detecta una colisión todas las estaciones dejarán de transmitir.

Una red CSMA/CD puede además incluir una política de retraso de retransmisión (*backoff squeme*). Sin esta política, todos los transmisores detectarían la colisión y cesarían de transmitir; pero una vez que sensaran silencio tratarían de transmitir de nueva cuenta, ocasionando una nueva colisión. La política de retraso implica hacer una decisión aleatoria, se retransmite o no, cuando se detecta silencio después de una colisión.

Un sistema CSMA/CD típico con esta política se conoce como retraso exponencial binario truncado (*truncated binary exponential backoff*). Cuando una estación está lista para transmitir y detecta silencio en la línea, enviará su mensaje con una probabilidad de 1 (100 %); esta probabilidad se conoce como persistencia. Si ocurre una colisión, la estación detendrá la transmisión y esperará a sensar silencio en la red. Cuando se detecta silencio en la red, la estación transmite con una

probabilidad de 0.5 (50 % de probabilidad de hacerlo y con 50 % de no hacerlo). Si se tienen dos estaciones involucradas en la colisión, ambas retroceden a 0.5 su condición de persistencia, existe un 50% de probabilidades de que una estación transmita y la otra espere a la siguiente oportunidad para transmitir, un 25 % de probabilidad de que ambas esperen a la siguiente oportunidad y 25 % de que ocurra de nuevo una colisión.

Si ocurre de nuevo una colisión, la persistencia se corta de nuevo a la mitad, es decir la persistencia será de 0.25. Cabe hacer notar que todas las estaciones involucradas en la colisión o colisiones disminuyen su condición de persistencia y que cada estación de forma independiente determina si retransmite o no.

La condición de persistencia es cortada a la mitad continuamente hasta que una estación logre transmitir o hasta que se generen 16 intentos fallidos de transmisión[1]. Si esto último ocurre la condición de persistencia regresa a uno y el proceso comienza de nuevo.

1.8.2 Probar y Seguir (Token Passing)

Regresemos al ejemplo del grupo de discusión sin moderador, pero añadamos una condición extra; ahora, para poder hablar la persona debe tener un micrófono. Todos los participantes escucharán, lo que la persona con el micrófono tenga que decir. Nadie que no tenga el micrófono puede transmitir. Una vez que la persona a terminado de hablar pasa el micrófono a la siguiente persona y así sucesivamente, si la persona que posee el micrófono no tiene nada que decir entonces se

[1] A pesar que una estación pueda experimentar 16 colisiones, la probabilidad de transmisión nunca será menor que 1/1024, ya que Ethernet y el standard IEEE 802.3 no permiten más de 1024 dispositivos en la red. A esto se debe el término de truncado en el nombre del esquema.

limita a pasarlo a la siguiente persona. Eventualmente la persona que inicio la transmisión tendrá el micrófono de vuelta y podrá hablar de nuevo si así lo desea.

El esquema descrito anteriormente es el conocido como probar y seguir (Token Passing). Este es el esquema que forma la base del sistema IBM conocido como Token Ring y representa el segundo algoritmo de control más usado en las redes de área local. El algoritmo de probar y seguir (token passing) es la base de los estándares 802.4 y 802.5 así como de las FDDI.

Cuando una estación tiene que enviar información a otra, necesita esperar a recibir el patrón de bits que representan el token. Los mensajes de token se envían de tal forma que sólo una estación lo tiene en un tiempo determinado. Por lo tanto, si una estación posee el token entonces es temporalmente la dueña de la red.

Si una estación recibe el token y no tiene nada que enviar simplemente transmite el token a la siguiente estación. Si tiene que enviar información, entonces genera un marco que la contiene. Después de enviar el marco con la información envía el token.

Una red token ring es un anillo lógico implementado en una topología física que soporte transmisión serial. Cada estación recibe un bit a la vez y regenera los bits para la estación siguiente. Cuando una estación termina de transmitir, transfiere el control a la estación siguiente enviándole el token.

Las redes token bus son conceptualmente similares a las redes token ring, excepto porque están implementadas usando una topología de transmisión simultánea (un bus por ejemplo), como se muestra en la figura 1.11. En esta topología física todas las estaciones escuchan todas las transmisiones. La estación transmisora direccionará el token a la siguiente estación del anillo lógico; aunque todas las estaciones puedan escuchar la transmisión, sólo la estación a la que está destinada la información podrá

hacer uso de ella. Después de recibir el token, la estación puede transmitir o bien enviar el token a la siguiente estación del anillo lógico. Eventualmente el token regresará a la primera estación cerrando el anillo.

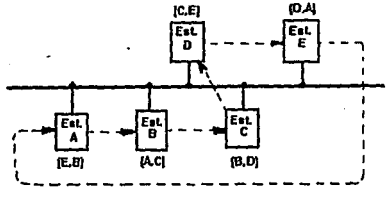


Figura 1.11

Bus Físico en Configuración Lógica de Anillo.

1.9 Arquitectura Cliente/Despachador

La arquitectura cliente/despachador es una forma de dividir e trabajo en diferentes computadoras. Las aplicaciones reciden en la máquina despachadora, la cual se encarga de responder a las peticiones de las máquinas clientes. El funcionamiento de esta arquitectura es aproximadamente el siguiente: un programa cliente requiere de ciertos procesos que no sabe ejecutar, pero sabe que existe un programa despachador que es capaz de brindar este servicio. La arquitectura cliente/despachador sirve para aprovechar las grandes capacidades de algunas máquinas, las máquinas clientes son por lo

regular máquinas con reducidas capacidades de procesamiento (computadoras personales o terminales X), mientras que los despachadores son máquinas con capacidades de procesamiento elevedas (estaciones de trabajo, computadoras personales o super-computadoras).

La modularidad proporcionada por esta arquitectura permite distribuir de una forma efectiva el trabajo entre las diferentes computadoras de una red. Las aplicaciones de alta interacción con el usuario residen en el cliente, mientras que las aplicaciones que necesiten ser compartidas pueden residir en los despachadores. Los datos locales almacenados en el cliente, le dan al usuario cierta autonomía y los datos en los despachadores ayudan a reducir la responsabilidades de los usuarios al mismos tiempo que están dsiponibles para más de una sola persona.

1.10 Sistema Abierto

En los primeros años de la computación, cuando los sistemas informáticos eran sistemas centralizados, era común tener lo que se conoce como sistemas propietarios es decir completamente diseñados por un solo fabricante. Con el paso del tiempo se tuvo la necesidad de implementar sistemas formados por equipos de varios fabricantes trabajando de forma conjunta, esto es lo que conocemos como sistema abierto. La evolución hacia los sistemas abiertos se debió a las siguientes causas:

a) La aparición de las computadoras personales trajo como consecuencia la individualización de los sistemas de cómputo. Esto a su vez genera una ampliación del mercado, por lo que surgieron muchas compañías dedicadas a desarrollarlos.

b) Tradicionalmente las empresas tendían al uso de sistemas propietarios, lo que las encadenaba al fabricante de su sistema o a los fabricantes que lo emularan. Por lo tanto la construcción de innovaciones dependían de la capacidad de su fabricante.

c) Los campos de aplicación de los sistemas de cómputo a crecido a tal grado que es prácticamente imposible generar un sistema que sea eficiente en los diferentes campos al mismo tiempo.

d) Cuando los usuarios de los sistemas de cómputo personales se dan cuenta de las grandes ventajas de la interconexión de los mismos ya existen muchas compañías que los fabrican, y en una misma oficina se encuentran equipos de marcas diferentes.

La aceptación de los estándares fue un hecho clave para la creación de los sistemas abiertos, sin éstos, no sería posible que sistemas de diferentes fabricantes se comporten como un sistema homogéneo. El objetivo de los sistemas abiertos es el generar aplicaciones: fáciles de portar de un sistema a otro, escalables y que sean inter-operables mediante el uso de los estándares.

El uso de los sistemas abiertos permite a los usuarios determinar la mejor solución a sus necesidades en base a la tecnología disponible y no simplemente por un logotipo o marca. Además los sistemas abiertos facilitan la construcción de nuevos sistemas que coexistan con los antiguos sistemas ya implementados.

Capítulo II

Principios de Telecomunicaciones

En el capítulo anterior se estudiaron las redes como un sistema, y para poder hacerlo de forma efectiva hicimos una división entre los aspectos de comunicaciones y los aspectos de las aplicaciones. En el presente capítulo se estudiará la parte correspondiente a la sub_red de comunicaciones. Como la mayoría de los adelantos tecnológicos, los sistemas de cómputo y las redes mismas están cambiando constantemente por lo tanto el conocimiento de las formas básicas de comunicación entre las diferentes máquinas es de suma importancia para la construcción de la red y para el estudio de futuros cambios en la misma.

La construcción de las redes no se limita a conectar varios equipos y compartir información entre ellos (aunque se espera que algún día pueda ser así), para tener una red funcionando de forma adecuada es necesario estudiar la forma física en que será manejada la información. Por ejemplo, el ancho de banda del medio de transmisión determinara la máxima y mínima velocidades de transmisión a través de la red, así mismo de acuerdo al tipo de información transmitida (datos, texto, voz, imágenes, etc.) y de la cobertura de la red, será más práctico utilizar un medio de transmisión que otro. Por otra parte el sistema de comunicación (hardware) debe ser escogido de tal forma que sea redituable económicamente y práctico, dando prioridad a la planeación para la extensión de la red.

Para construir sistemas de gran cobertura se decició usar la red telefónica y de esta forma no tener que establecer una nueva vía de comunicación.

La estrecha relación entre los sistemas telefónico y de red a hecho que los avances realizados en alguno de los sistemas repercuta en el otro , de esta forma las lineas telefónicas pronto empezaron a digitalizarse. La generación de los nuevos sistemas telefónicos han hecho realidad una red de comunicación que permite transmisión de diferentes tipos de información, voz, datos, etc. Al sistema de comunicación se le conoce como Red Digital de Servicios Integrados (Integraded Services Digital Network ISND). Este tipo de sistemas fue implementado originalmente por la compañía telefónica AT&T. Al privatizarse el sistema telefónico mexicano, la compañía Teléfonos de México tiene que digitalizar sus sistemas, para poder competir con las empresas de telefonía extranjeras que se espera lleguen como consecuencia de la firma del Tratado de Libre Comercio.

2.1 Codificación de la información

Tanto en los sistemas de comunicación como en los de procesamiento de datos, la información tiene que ser codificada de alguna manera para ser tratada. El gran mérito de la clave Morse es el haber sentado las bases para el diseño de un código binario (en base a dos símbolos solamente) para la transmisión de información mediante una máquina. En términos generales esta idea es usada inclusive en los sistemas modernos de comunicación y de procesamiento. Los datos se construyen mediante una unidad fundamental conocida como *bit* (Binary digiT), la cual sólo puede tener dos valores (por lo que decimos que es un código binario) cero o uno. Sin embargo la forma más simple de comunicación entre computadoras y seres humanos es mediante arreglos de caracteres que forman un texto. Para poder establecer una relación entre los bits, forma de información que maneja la computadora, y los caracteres, forma de comunicación utilizada por los seres humanos, cada carácter tiene asociado una combinación única de ceros y unos (bits) la cual lo identifica. En la actualidad se utilizan otros tipos de comunicación como: gráficas, sonido, señales como el puntero del ratón, etc.; pero, el texto sigue siendo la forma más popular de comunicación.

Existen diferentes códigos encargados de establecer la correspondencia entre bits y caracteres. Todos los códigos se basan en el mismo principio, existe un predeterminado grupo de bits dentro del cual cada diferente combinación de unos y ceros representa un carácter. Al conjunto de bits utilizado para formar el código se le conoce como "palabra" (word). El uso del término "palabra", como muchos otros términos dentro de los sistemas digitales, puede prestarse a confusión gracias al uso indiscriminado que se hace de ellos.

Como los primeros procesadores eran de ocho bits, los fabricantes de procesadores definieron la "palabra" como un grupo de ocho bits; más tarde, con el advenimiento de los procesadores de dieciséis y treinta y dos bits, estos mismos fabricantes redefinieron el término "palabra", primero como un grupo de dieciséis bits y luego como uno de treinta y dos. Es por esto que nosotros utilizamos el término de "palabra" en su sentido más general, es decir, una palabra es un grupo predefinido de bits que indican el tamaño de la unidad de información.

Los códigos más utilizados para definir carácteres son:

-El código Baudot que debe su nombre a su autor, uno de los primeros códigos desarrollados

-El código ASCII, siglas de su nombre en inglés, American Standard Code for Information Interchange, el cual es estándar internacional

-El código EBCDIC también siglas de su nombre en inglés, Extended Binary Coded Decimal Interchange Code, que intento suplantar al código ASCII como estándar internacional.

El código Baudot se construye a base de una palabra de 5 bits, es decir cada diferente quintupla de bits representa un carácter diferente. El usar una palabra de cinco bits limita a treinta y dos el número de caracteres que pueden ser representados mediante este código ($2^5 = 32$ caracteres). Esto resulta insuficiente inclusive para representar las veintiséis letras del alfabeto inglés y los diez caracteres numéricos. Para resolver este problema se diseño un mecanismo que permitió aumentar a sesenta y cuatro el número de posibles combinaciones, sin necesidad de aumentar el largo de la palabra. El mecanismo consistía en generar dos tablas para que cada diferente quintupla tenga dos posibles representaciones, una de acuerdo a cada una de las tablas.

En cada tabla, se usa una de las combinaciones para generar un carácter especial, sin representación gráfica, que sirve para conmutar de una tabla a otra. Los caracteres de control se conocen como "cambio a letras" (letter shift) y "cambio a números" (number shift). Una de las ventajas del código Baudot es su tamaño, al usar una palabra corta, de 5 bits, la transmisión se hace más rápida. Su desventaja es la de tener que estar conmutando constantemente de una tabla a otra, durante la escritura de un texto.

El código Baudot fue utilizado en Europa como código de transmisión de los teletipos y telex. Estos mismos dispositivos en América utilizaron un código distinto, el código americano estándar para intercambio de información (ASCII). El código ASCII es un código de siete bits. El largo de palabra de este código, le permite ciento veintiocho posibles combinaciones ($2^7 = 128$ caracteres), esto elimina la necesidad de conmutar entre dos tablas de caracteres. El código ASCII a sido expandido a ocho bits para poder representar caracteres matemáticos y de gráficos, al código ASCII de ocho bits se le conoce como ASCII extendido.

Además de ser utilizado para comunicaciones el código ASCII se usa como código estándar en las computadoras personales, es por esto que además de representar todos los caracteres alfanuméricos incluye signos de puntuación, caracteres de gráficos y caracteres de control de enlace de datos. Los caracteres de gráficos son los que ayudan a definir como aparecerán los caracteres en la pantalla o en la impresora. Entre los caracteres de gráficos se incluye el retorno de carro, la alimentación de línea, tabuladores, etc. Los caracteres de enlace de datos, brindan información a cerca de la interpretación del mensaje en sí mismo. Ejemplos de este tipo de caracteres son: inicio del texto (Start of TeXt STX), fin de transmisión (End Of Transmition EOT), control de dispositivo (Device Control), etc.

Un tercer código fue desarrollado por la compañía Máquinas de Negocios Internacionales (International Business Machines IBM), para ser usado con sus mainframes. El código desarrollado por la IBM recibió el nombre de código binario extendido para el intercambio de codificación decimal (Extended Binary Coded Decimal Interchange Code EBCDIC). Este es un código de ocho caracteres, lo que permite doscientas cincuenta y seis combinaciones; sin embargo, algunas combinaciones no tienen representación. Estos huecos en el código lo hacen difícil de utilizar, quizá es por esto que el código ASCII predominó como el código más popular.

2.2 Formas de transmisión

Existen dos formas de transmisión, serie y paralelo. La transmisión en paralelo consiste en enviar un grupo de bits al mismo tiempo. De las dos formas de transmisión esta es la más rápida, ya que se permite disponer de un grupo de bits en un mismo instante; sin embargo, presenta la desventaja de usar una vía de comunicación para cada uno de los bits que forman el grupo, lo que la hace prohibitiva para la transmisión a grandes distancias. Para la transmisión serial, los datos se envían uno tras otro, como usa la misma vía de transmisión para todos los bits, es altamente recomendable para la transmisión de información a grandes distancias. Su desventaja principal es que es mucho más lenta que la transmisión en paralelo.

La transmisión serial se divide a su vez en dos formas más, síncrona y asíncrona. La transmisión síncrona, necesita una señal de reloj entre la unidad transmisora y la receptora para coordinar la transmisión de la información con su recepción. La unidad receptora tiene los circuitos de recuperación de reloj (Clock Recovery Circuits) cuya función es la de extraer la señal del reloj del flujo de información.

La sincronización se establece cuando la unidad receptora extrae la señal de reloj del flujo de datos. La sincronización implica que se ha establecido, tanto el principio como el fin del grupo de bits que forman la "palabra" (entiéndase *palabra* como unidad de transmisión, no en su sentido literal) para que los caracteres puedan ser descodificados correctamente. Subsecuentemente la sincronización se mantiene mediante la señal de reloj.

La transmisión de información en forma asíncrona usa bits extra, bits de marco, (framing bits) para establecer el inicio y fin de cada bloque de transmisión. El receptor se prepara a recibir un carácter cuando detecta el bit de inicio (start bit). El flujo de datos empieza a ser decodificado hasta que el receptor recibe el bit de paro (stop bit). Aunque no se utilice la señal de reloj para controlar la transmisión es necesario que la unidad receptora y la transmisora manejen la misma frecuencia de reloj para que la información sea muestreada a la misma frecuencia con que es enviada. La velocidad de muestreo establece la sincronización de bits mientras que los bits de comienzo (Star bit) y el de final (Stop bit) establecen la sincronización de caracteres.

Como la cantidad de bits de marco (framing bits) depende del equipo utilizado es necesario medir la eficiencia del sistema de transmisión. La eficiencia de un sistema digital de transmisión es:

$$eficiencia = \frac{Bits\,de\,datos}{Total\,de\,bits\,transmitidos} x\,100\%$$

A simple vista la transmisión en forma síncrona parece ser mucho mejor que la asíncrona; sin embargo, para determinar la forma de transmisión para un sistema es necesario tomar encuentra otros factores y no sólo la eficiencia de la forma de transmisión.

2.3 Sistema básico de comunicación

El sistema de comunicación más sencillo es el constituido por una sola entidad transmisora y sólo una entidad receptora. Cuando la conexión entre el receptor y el transmisor no tiene conexiones intermedia, más que amplificadores o repetidores, decimos que tenemos un enlace directo. Si existe enlace directo entre el transmisor y el receptor y estos son los dos únicos equipos compartiendo el medio de transmisión se dice que tenemos un enlace punto a punto. Cuando existen varios dispositivos compartiendo el medio de transmisión tenemos un enlace multipunto.

De esta forma un sistema básico de comunicación consiste en sólo una unidad transmisora y una sola receptora con un enlace punto a punto. Esta pequeña red comienza del lado izquierdo con la estación primaria (transmisora). La información es enviada de esta estación a una estación remota o secundaria (receptora) en el otro lado del enlace. El sistema de comunicación se divide básicamente en dos partes, el equipo terminal de datos (Data Terminal Equipment, DTE) y el equipo de comunicación de datos (Data Comunicactions Equipment DCE) como puede verse en la figura 1.1, en al capítulo anterior.

La sección de comunicaciones del DTE incluye un procesador de programas de aplicación, la unidad de control de línea (Line Control Unit LCU) y el "receptor / transmisor" universal asíncrono (Universal Asynchronous Reciver Transmitter UART) o el "receptor / transmisor" universal síncrono asíncrono (Universal Sincrhonous Asinchronous Reciver transmitter USART). En la figura 2.1 se puede observar como la sección de comunicación está conectada con los periféricos a través de la LCU.

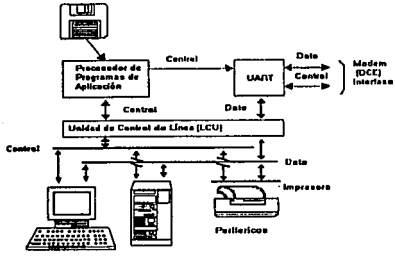


Figura 2.1 Equipo Terminal de Datos (DTE)

Como el manejo de la información en paralelo es más rápido, por lo general la información se procesa en paralelo y se transmite en serie. Los dispositivos antes mencionados, el UART y el USART son los encargados de convertir la información en paralelo a información serial y viceversa. El proceso de conversión y la velocidad a la cual se envían los datos paralelos hacia el UART o el USART y la velocidad en que se envían los datos seriales a través del medio de transmisión son manejados por el sistema de computadora al cual esté conectado el UART o USART.

La unidad de control de línea (LCU) es un dispositivo localizado generalmente como parte de la entidad transmisora, la LCU da acceso a la unidad transmisora a los datos enviados por los periféricos, además contiene el software necesario para establecer y controlar la transmisión de datos entre el receptor y el transmisor (protocolo).

El protocolo tienen la función de seleccionar el orden de acceso de los diferentes periféricos conectados a la LCU, establecer el formato a usarse para interpretar el flujo de datos enviado por los periféricos y convertirlos en información útil: señales de control, mensajes, señales para detección de errores, etc. Los programas de aplicación también son capaces de enviar señales de control para permitir el flujo de información de los periféricos, atendidos por el UART o el USART , hacia el equipo de comunicación de datos (Data Communications Equipment DCE).

Los dispositivos de comunicación de datos (DCE) tienen la función de convertir los datos digitales recibidos de los DTE's y convertirlos en señales capaces de ser enviadas a través del medio de transmisión, o bien efectuar la función inversa, recibir la información del medio de transmisión y convertirla en datos que puedan ser utilizados por el DTE. El equipo de comunicación de datos más común es el "modulador-demodulador", modem (MOdulator / DEModulator MODEM). Su función es la de convertir la información serial, obtenida del UART o del USART, en diferentes tonos de audio que pueden ser transmitidos a través de las líneas telefónicas. Existen muchos tipos de Modems, se clasifican de acuerdo a la cantidad y tipo de información que pueden manejar y a la forma en que la información es transmitida y/o recibida.

Para poder enviar la señal, la información debe ser codificada en algún formato digital (los formatos digitales se estudiarán más adelante), el dispositivo encargado de codificar la información es el codificador / decodificador (COder/DECoder CODEC).

Otra parte importante del sistema de comunicación es sin lugar a dudas el medio de transmisión, por lo tanto a esta parte le dedicaremos un apartado más adelante en este capítulo.

Finalmente en la parte en la unidad receptora se encuentra el controlador de estación (STAtion COntroler STACO), contraparte de la LCU en la unidad receptora. La STACO, en lugar de controlar el enlace de la información, responde a los comandos y reglas del protocolo para poder dirigir la información al periférico correcto y enviar la información que le sea requerida como respuesta.

2.4 Enlaces de comunicación

Los enlaces de comunicación están diseñados para satisfacer los requerimientos de algún sistema en particular. Como ya hemos mencionado, el enlace más sencillo es aquel que contiene una estación transmisora que es la encargada de establecer la comunicación y controlarla mientras que la otra estación, la receptora, tan sólo recibe la información y contesta a los requerimientos de la primera. Las dos estaciones deben usar el mismo protocolo, tener la misma capacidad para el manejo de datos y tener el mismo código de datos para establecer un buen enlace. Los métodos actuales de enviar y recibir información están divididos en tres unidireccional (Simplex), semi-bidireccional (Half duplex), y completamente bidireccional (Full duplex).

Un sistema configurado para establecer comunicación sólo en un sentido, se considera como un sistema con transmisión unidireccional (simplex). La estación transmisora siempre es la que envia información a la estación receptora sin requerir de esta última ninguna señal de respuesta. Este tipo de enlace es útil cuando es necesario transmitir grandes cantidades de información que no necesita de confirmación de recepción. Cuando se utiliza este tipo de enlace, se utiliza una forma de comunicación secundaria para confirmar la llegada de la información a su destino.

Los enlaces semi-bidireccionales permiten la transmisión de información en ambos sentidos, del transmisor al receptor y viceversa, pero restringe esta transferencia a sólo un sentido a la vez. Hasta que la estación emisora haya terminado de enviar su mensaje la estación receptora podrá enviar la respuesta. Los enlaces completamente bidireccionales permiten la transmisión simultánea de información en ambos sentidos. Una estación A envía un mensaje hacia una B usando un par de tonos de audio (cada uno con su respectiva frecuencia) para representar unos y ceros. Al mismo tiempo, la estación B transmite información hacia la estación A usando también un par de tonos de audio, sólo que diferentes a los usados por la estación A. La diferencia entre los pares de tonos permite que las dos estaciones utilicen el mismo medio de transmisión al mismo tiempo para enviar sus mensajes.

2.5 Frecuencia, espectro y ancho de banda

En el ambiente de las comunicaciones a la información que se transmite se le conoce como señal. La señal no es otra cosa que una función capaz de representar la variación de un parámetro a través del tiempo, este parámetro puede ser un voltaje una corriente, etc. Como la mayoría de las señales manejadas en comunicaciones son periódicas, tenemos que la señal puede ser representada como una función del tiempo o de la frecuencia.

En el dominio del tiempo uno de los conceptos más importantes es la continuidad de la función. Una señal es continua si:

$$\lim_{t \to a} f(t) = f(a) \quad \forall \ a$$

Una señal es discreta si sólo está definida para un número finito de valores. La parte (a) de la figura 2.2 muestra una señal continua mientras la figura 2.2 (b) presenta una señal discreta.

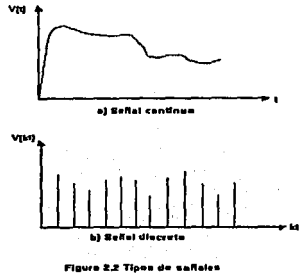


Figura 2.2 Tipos de señales

Otro aspecto interesante de la señal es la periodicidad. Una señal es periódica si y sólo si:

$$f(t+T) = f(t) \qquad \text{para: } -\infty < t < \infty$$

donde la constante T es el periodo de la señal (T es el valor más pequeño que satisfaga la ecuación). Las tres características más importantes de una señal periódica son: la amplitud, la frecuencia y la fase. La amplitud es el valor instantáneo de la señal en cualquier punto. La frecuencia es el valor inverso del periodo (1/T), o el número de repeticiones del período en un segundo; la frecuencia se expresa en ciclos por segundo, o hertz (Hz). La fase es una medida de la posición relativa de la señal dentro del período de la misma. En la figura 2.3 muestra dos señales desfasadas 90° o $\pi/2$ radianes (2π radianes = 360°).

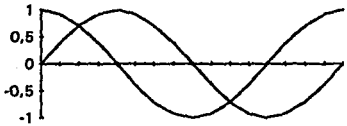


figura 2.3 señal desfasada

Hasta ahora hemos estudiado la señal como función del tiempo, pero como ya hemos mencionado también podemos considerarla como función de la frecuencia. Por ejemplo, considere la función: $f(t) = sin(2\pi f_1 t) + sin(2\pi(3f_1)t)$ (cuya gráfica se muestra en la figura 2.4) se forma por la suma de ondas senoidales de frecuencias f_1 y $3f_1$, las partes (a) y (b) de la figura muestran las componentes individuales, en las cuales se puede observar que: la segunda frecuencia es un múltiplo de la primera (frecuencia fundamental) y que el período total de la señal es igual al período de la frecuencia fundamental. El período de la componente $sin(2\pi f_1 t)$ es: $T = 1/f_1$ y el periodo de $f(t)$ es también T. Tal como se muestra en la figura 2.4 (c).

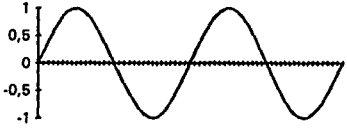


figura 2.4 (a)

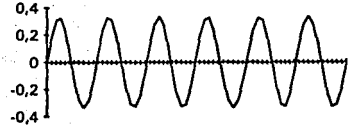


figura 2.4 (b)

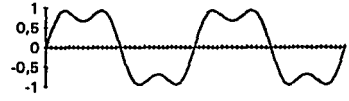


figura 2.4 (c)

figura 2.4 Suma de componentes de frecuencia

Por lo tanto podemos decir que para cada señal existe una función en el dominio del tiempo *f(t)* que especifica la amplitud de la señal en cada instante y una función en el dominio de la frecuencia que muestra las frecuencias de las cuales está constituida dicha señal. La figura 2.5(a) muestra la gráfica de la función en el dominio de la frecuencia *f(s)* de la señal mostrada en la figura 2.4(c). En la parte (b) de la figura 2.4 se muestra la función, en el dominio de la frecuencia, de una onda cuadrada que vale uno entre -x/2 y x/2, y cero en cualquier otro lugar. En este caso la función *f(s)* es continua y nunca llega a cero, aunque la amplitud de los diferentes componentes de frecuencia se reduzca conforme la frecuencia se hace más grande. Este comportamiento es común en las señales reales.

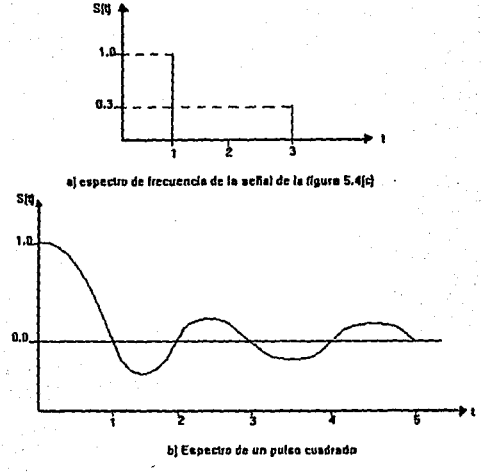


figura 2.5 Señales en el dominio de la frecuencia

El espectro de la señal es el rango de frecuencias que contiene. Para la señal de la figura 2.4 (c), el espectro se extiende desde f_1 hasta $3f_1$. Como el ancho de banda absoluto de la señal es el ancho de banda del espectro, el ancho de banda de esta señal es $2f_1$. Por el contrario muchas señales, como la mostrada en la figura 2.5 (b), tienen un ancho de banda infinito.

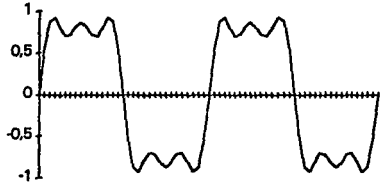
Cuando una señal incluye una componente de frecuencia cero, este componente recibe el nombre de componente de corriente directa o constante. La figura 2.6 muestra el resultado de añadir una componente de corriente directa a la señal de la figura 2.4(c). Sin componentes de corriente directa la señal tiene una amplitud promedio de cero, como puede observarse de la gráfica en el dominio del tiempo. Cuando se le añade una componente de corriente directa, tiene una componente de frecuencia igual a cero y una amplitud promedio diferente de cero.

2.6 Relación entre velocidad de transmisión y ancho de banda

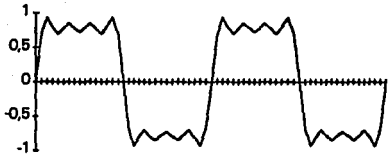
En términos generales podemos decir que el ancho de banda es la región dentro de la cual se encuentra la mayor parte de la energía de la señal. La cuestión más importante aquí, es que a pesar que la señal, en teoría puede contener un rango infinito de frecuencias; en la práctica, sin importar el medio de transmisión que se use, sólo se podrá manejar un rango limitado de estas frecuencias. Esto limita la velocidad de transmisión que puede ser usada en un medio determinado. Para tratar de explicar esta relación, considere una señal cuadrada, como la obtenida a la salida de un oscilador digital, tome el pulso positivo como uno lógico y el pulso negativo como cero lógico. Entonces la señal reprsenta un flujo de ceros y unos. La duración de cada pulso es $0.5 f_1$; de esta forma, la velocidad de transmisión es de $2 f_1$ bits por segundo (bps).

Como puede observarse de la figura 2.5(c) al sumarse las dos señales, la de frecuencia f_1 y la de frecuencia $3 f_1$, la señal resultante tiende a simular una señal cuadrada. Si continuamos e proceso de añadiendo una señal de frecuencia $5 f_1$, se obtiene la señal de la figura 2.7(a), y añadiendo a esta una señal de frecuencia $7 f_1$ obtenemos la señal mostrada en la figura 2.7(b), podemos continuar el proceso hasta el infinito para obtener la señal de la figura 2.7(c).

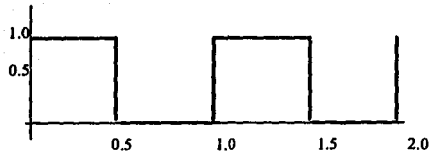
figura 2.6 señal alterna con una componente de directa



a) $sin(2\pi f_1 t)+\frac{1}{3}sin(2\pi(3f_1)t)+\frac{1}{5}sin(2\pi(5f_1)t)$



b) $sin(2\pi f_1 t)+\frac{1}{3}sin(2\pi(3f_1)t)+\frac{1}{5}sin(2\pi(5f_1)t)+\frac{1}{7}sin(2\pi(7f_1)t)$



c) $\sum \frac{1}{k} sin(2\pi(k f_1)t)$

figura 2.7 Componentes de frecuencia de una señal cuadrada

Conforme vamos añadiendo componentes de frecuencia que sean múltiplos nones de la frecuencia f_1 la señal resultante se aproxima más a una onda cuadrada. Esto se debe a que una señal cuadrada tiene un número infinito de componentes de frecuencia. Sin embargo, la amplitud de la k^a componente de frecuencia, kf_1, es solamente 1/k, por lo que podemos afirmar que la mayor parte de la energía se concentra en los primeros componentes de frecuencia.

Para entender mejor la relación entre ancho de banda y velocidad de transmisión, suponga un medio de transmisión digital que es capaz de transmitir señales en un ancho de banda de 4MHZ, por medio del cual intentamos transmitir una serie de una forma bastante eficaz a una señal cuadrada.

Si la frecuencia de la señal es de 1MHz, entonces el ancho de la señal:

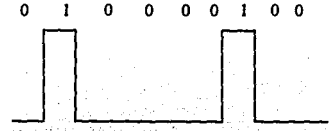
$$f(t) = sin(2\pi \times 10^6 t) + \frac{1}{3} sin(2\pi \times 3 \times 10^6 t) + \frac{1}{5} sin(2\pi \times 5 \times 10^6 t)$$

es de $(5\times10^6)-10^6=2$ MHz, lo que está dentro del ancho de banda del medio transmisor. Note que para una frecuencia f_1=1 MHz, el periodo de la frecuencia fundamental es de T=1μseg. De esta forma si consideramos esta forma de onda como un flujo de ceros y unos, los bits se presentan cada 0.5 μseg , lo que nos da una velocidad de transmisión de $2\times10^6=2$ Mbps. Por lo tanto, para un ancho de banda de 4 MHz, se tiene una máxima velocidad de transmisión de 2 Mbps. Ahora suponga que el medio tiene un ancho de banda de 8 MHz, además considere de nuevo la señal de la figura 2.7(a), pero suponga que tiene una frecuencia de 2 MHz. Usando un razonamiento similar al establecido en el análisis anterior, el ancho de banda de la señal es $(5\times2\times10^6)-(2\times10^6)=$ 8 MHz, de nuevo dentro del rango permitido. Pero en este caso, T=0.5μseg, por lo que se presenta un carácter cada 0.45μseg. Obteniéndose por resultado una velocidad de transmisión de 4Mbps. Lo que demuestra que si aumentamos al doble el ancho de banda, aumentamos también al doble la máxima velocidad de transmisión.

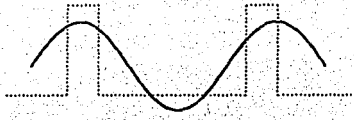
Pero eso no es todo, ahora considere la señal mostrada en la figura 2.4(c) como la señal aproximada a la onda senoidal. Si consideramos que esta señal tiene una frecuencia de 2MHz y siguiendo un razonamiento similar al anterior, el ancho de banda de la señal es de $(3\times2\times10^6)-(2\times10^6)=2$ Mhz. Pero en este caso T = 0.5 μseg. Por lo tanto tenemos un nuevo bit cada 0.45 μseg lo que da una velocidad de transmisión de 2Mbps. Esto demuestra que un determinado ancho de banda, puede soportar varios tipos de señales, la selección entre una y otra entonces dependerá de las caracteristicas del receptor.

A manera de conclusión podemos decir lo siguiente: En general, una onda puede tener un ancho de banda infinito. Si intentamos transmitir esta señal a través de un medio, la naturaleza del medio restringirá el posible ancho de banda de transmisión. Adicionalmente, sin importar el medio de transmisión, entre mayor sea su ancho de banda mayor es su costo. De esta forma, para poder ser transmitida de una forma económica y práctica, la señal digital tiene que ser aproximada por medio de una señal con un ancho de banda finito. Por otro lado, al aproximar la señal digital para limitar el ancho de banda, la señal se distorsiona, lo que genera problemas para interpretar la señal. Entre más limitado sea el ancho de banda mayor será la distorsión y por lo tanto mayor será el potencial de error en el receptor.

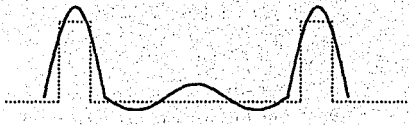
Daremos un ejemplo más que será útil para reforzar estos conceptos. La figura 2.8 muestra una señal digital con una velocidad de 200 bps. Con un ancho de banda de 1700 a 2500 Hz, la aproximación es bastante buena. Con un ancho de banda de 4000 Hz, la aproximación es muy buena. Podemos generalizar estos resultados de la siguiente forma: Si la velocidad de una señal es de W bps, entonces se puede obtener una muy buena aproximación de la señal con un ancho de banda de $2W$ Hz.



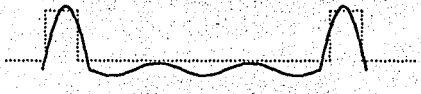
Pulsos antes de la transmisión 2000 bits por segundo



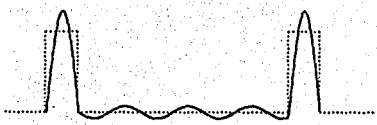
a) ancho de banda 500Hz



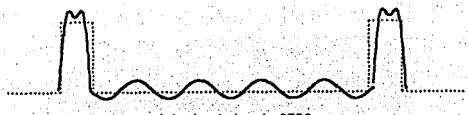
b) ancho de banda 900 Hz



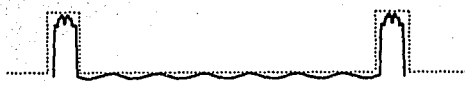
c) Acho de banda 1300 Hz



d) Ancho de banda 1700 Hz



e) Ancho de banda 2500



f) Ancho de banda 4000 Hz

2.8 Efecto del ancho de banda en una señal digital

Podemos decir que existe una relación directa entre la máxima velocidad de transmisión y el ancho de banda: entre más grande sea la máxima velocidad de transmisión, mayor debe ser el ancho de banda. O planteándolo en sentido inverso, entre mayor sea el ancho de banda mayor será la máxima velocidad de transmisión del sistema.

Otra observación que vale la pena hacer es ésta: Si pensamos el ancho de banda de una señal como centrado en alguna frecuencia, frecuencia central, entre mayor sea la frecuencia central, mayor será el ancho de banda potencial, y por lo tanto mayor será la máxima velocidad de transmisión. Si una señal está centrada en una frecuencia de 2000 Hz, su ancho de banda máximo es de 4000 Hz.

2.7 Intensidad de la señal

Otro parámetro importante en un sistema de transmisión es la intensidad de la señal a transmitir. Conforme la señal se propaga a través del medio de transmisión, sufre pérdidas o atenuaciones. Para compensar estas pérdidas se colocan amplificadores a lo largo de la línea de transmisión. La unidad que se utiliza para expresar la ganancia, pérdida y la magnitud de niveles relativos son los decibeles. El uso de los decibeles se debe a dos razones principalmente. La primera es que la intensidad de la señal por lo general cae de forma logaritmica, de esta forma las pérdidas se pueden expresar fácilmente en decibeles que son una unidad logaritmica. La segunda razón es que tanto la pérdida como la ganancia en un patrón de comunicación en cascada, puede obtenerse con operaciones simples de adición y substracción.

Básicamente un decibel, es la medida de la diferencia entre dos niveles de magnitud:

$$N_{dB} = 10 \log_{10} \frac{P_1}{P_2}$$

donde:

N_{dB} = número de decibeles

$P_{1,2}$ = magnitudes que se comparan

$\log_{10}$ = logaritmo base 10

Por ejemplo, si una señal de 10 mW se inserta dentro de una línea de transmisión y después de haber recorrido una distancia X su intensidad es de 5 mW, la pérdida puede ser expresada como:

Pérdida = 10 log (s/10) = 10 (-0.3) = - 3 dB

Cabe hacer notar que los decibeles son una medida relativa y no absoluta. Una pérdida de 1000 W a 500 W es también una pérdida de - 3dB. Por lo tanto una pérdida de 3dB significa una pérdida de la mitad de la magnitud de la señal, y una ganancia de 3dB significa que la intensidad de la señal subió al doble de su valor.

2.8 Transmisión digital y transmisión analógica

En la transmisión de datos, se debe estar consciente del tipo de información que se está manejando, para poder determinar el medio de transmisión más apropiado y los dispositivos y técnicas necesarios para manejar la información de manera que llegue a su destino de forma inteligible. La información a transmitir puede ser digital o analógica, de forma burda podemos decir que una señal analógica es una señal continua y que una señal discreta representa una señal digital.

Ambos términos, digital y analógico, se pueden encontrar dentro de tres diferentes contextos: datos, señales y transmisión. Los datos son entidades que tienen un significado inherente. La diferencia entre datos e información es que los datos se refieren a la forma de algo, mientras que la información se refiere a la interpretación de esos datos. Las señales son formas eléctricas o electromagnéticas de codificar información. La señalización es el acto de propagar la señal a través de un medio. Finalmente, transmisión es la comunicación de datos mediante la propagación y el procesamiento de señales. En lo que resta de esta sección trataremos examinar los términos digital y analógico en estos tres contextos.

Los datos analógicos toman valores continuos durante algún intervalo mientras que los datos digitales sólo pueden tomar valores discretos. En un sistema de comunicaciones, los datos se propagan de un punto a otro en forma de señales eléctricas. Una señal analógica es una onda electromagnética que se propaga a través de un medio. Una señal digital es una secuencia de pulsos de voltaje que también son transmitidos a través de un medio. Las principales ventajas de la señalización digital es que, en general, es más barata que la analógica y que es más resistente a la interferencia. Su principal desventaja es que las señales digitales sufren mayor atenuación que las analógicas.

Tanto las señales digitales como las analógicas son susceptibles de ser transmitidas a través de un medio. La forma en que estas señales son tratadas está en función del sistema de transmisión. La transmisión analógica se refiere al efecto de enviar señales analógicas sin considerar su contenido, las señales pueden ser datos analógicos (voz, por ejemplo) o datos (información enviada a través de un modem). Conforme aumenta la distancia entre el emisor y el receptor, la señal sufre mayor atenuación. Para aumentar la potencia de la señal enviada se utilizan amplificadores; sin embargo, además de aumentar la señal transmitida, también amplian el ruido. Para datos analógicos se puede permitir un poco de distorsión por ruido, pero para señales digitales el aumento de la señal de ruido puede generar graves errores. Es por esto que para señales digitales es más conveniente el uso de repetidores. Como su nombre lo indica un repetidor, reconoce una secuencia de bits y luego la retransmite. Los repetidores también pueden ser utilizados con señales analógicas para tener señales más limpias en los receptores.

Para establecer comunicación a largas distancias, no es recomendable el uso de señales digitales; sin embargo, la transmisión digital es superior a la analógica en términos de costo y calidad, por lo que los sistemas de comunicación de amplio alcance tratan de digitalizar en la medida de lo posible su forma de transmisión.

2.9 Formatos binarios de señales digitales

Además de los diferentes códigos de caracteres y formas de transmisión (síncrona y asíncrona), la información digital puede ser codificada en diferentes formatos de señales eléctricas. Los formatos de señal más comunes se muestran en la figura 2.9. Cada formato de señal, es enviado como un flujo de datos, y puede ser interpretado como una onda cuadrada cuya frecuencia varía de acuerdo al patrón de bits que se transmite. Generalmente la frecuencia de la señal cuadrada se reduce conforme crece el número de ceros o unos consecutivos aumenta. Esto no es válido para todos los formatos de señal, como veremos más adelante, pero sirve como una primera aproximación al significado de la frecuencia de la onda senoidal fundamental (la frecuencia fundamental de la señal). Suponga que se tiene una serie de bits, consistente en unos y ceros intercalados (...1010101...). En un esquema esto aparecería como una señal que constantemente conmuta entre dos niveles. En esencia la señal aparecerá como una señal cuadrada con una frecuencia determinada. Ahora suponga que la señal consiste en un flujo de bits donde ahora los ceros y unos se intercalan por pares (...1100110011...). La señal generada por este flujo de bits tiene la mitad de la frecuencia de la anterior, ya que su período aumentó al doble. La importancia de la onda fundamental es que la potencia máxima de la onda cuadrada se encuentra concentrada en ella.

En la figura 2.9 se muestra la onda fundamental de la señal cuadrada mediante una línea punteada, en la porción del flujo de datos que la contiene. Como puede observarse de la figura las otras combinaciones de ceros y unos la frecuencia de las ondas generadas o bien es la misma o es menor que la fundamental. A continuación se describen los formatos mostrados en la figura 2.9.

2.9.1 Señal sin retorno a cero (NRZ)

La señal sin retorno a cero (Non Return to Zero), es el formato binario de señal clásico. El uno lógico tiene asociado un valor de voltaje positivo (+V) y el cero está asociado al cero volts o tierra (0V). Esta es la forma en que la mayoría de los estudiantes de electrónica conocen las señales digitales. En este formato de señal se cumple que la frecuencia fundamental ocurre cuando se transmiten unos y ceros alternados. La señal fundamental señala la máxima velocidad de cambio que podrá tener la señal en un momento dado, por lo tanto el medio de transmisión debe tener un ancho de banda lo suficientemente bueno como para responder a esta velocidad de cambio. Este es el formato de señal más fácil de producir, ya que sólo requiere de un dispositivo capaz de generar o un uno (encendido) o un cero (apagado).

2.9.2 Señal sin retorno a cero bipolar (NRZB)

Este es un formato de señal muy similar al anterior, por lo tanto no se ilustra en la figura 2.9. La diferencia con el anterior radica en que éste en lugar de asignar el valor de tierra para el cero lógico, se le asigna un voltaje negativo (-V). Este tipo de señal se utiliza para escribir en medios magnéticos.

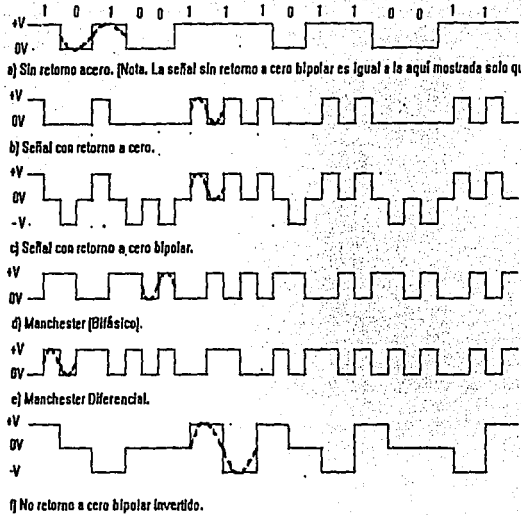


Figura 2.9 Formato Binario de Datos.

2.9.3 Señal con retorno a cero (RZ)

La señal con retorno a cero (Return to Zero), usa cero volts (0V) para definir el cero lógico y usa un voltaje positivo (+V) para representar el uno lógico. Este formato de señal; sin embargo, hace que la señal de uno lógico regrese a cero en la mitad del tiempo que debería estar presente. El regreso a cero de la señal sirve para prevenir que la señal se mantenga en uno lógico por largos periodos de tiempo, en caso que se transmita una serie larga de unos consecutivos.

La frecuencia fundamental ocurre cuando se transmiten dos unos consecutivos, ya que en la mitad del tiempo de vida del bit la señal bajará a cero y después tendrá que subir para transmitir el siguiente uno. Esto tiene como resultado una relación uno a uno entre la máxima frecuencia de transmisión y la frecuencia fundamental. Lo que exige un medio de transmisión con un ancho de banda mayor que el que se requeriría para transmitir una señal sin retorno a cero a la misma velocidad.

2.9.4 Señal con retorno a cero bipolar (RZB)

La señal con retorno a cero bipolar (Return ti Zero Bipolar), como la NRZB asocia un voltaje positivo (+V) al uno lógico y un voltaje negativo (-V) al cero lógico. La diferencia con la NRZB es que a la mitad del tiempo de cada bit la señal regresa a cero. De esta forma no sólo provee la característica de voltajes opuestos, sino que también presenta el cambio de nivel de la señal en cada periodo. Los sistemas síncronos, encuentran ventajoso este formato para la recuperación de las señales de reloj.

2.9.5 Codificación Manchester

El formato de señal Manchester es un sistema de codificación de señal más que un formato. La cadena de datos es proporcionada por un circuito que complementa (invierte) la primera mitad del bit de dato. La siguiente mitad se deja con su valor. El propósito de esto es crear una transición regular en el centro de cada bit para ayudar a la recuperación de la señal de reloj en transmisiones síncronas.

2.9.6 Codificación Manchester diferencial

Como en el código Manchester cada bit presenta una transición a la mitad de su periodo. Para formar el código diferencial los niveles lógicos se forman mediante la comparación con el siguiente bit. Si el segundo bit de la comparación es un uno lógico, el nivel de la primera mitad del período es igual al que tenga la primera mitad. Por el contrario si el siguiente bit de la comparación es un cero lógico la primera mitad del periodo del dato es el completo de la segunda mitad.

2.9.7 Señal sin retorno a cero con inversión de marco

En este formato de señal, el nivel de cero está reservado para los bits con valor de cero lógico. El nivel para el uno lógico los bits aparecen como niveles de voltajes alternados. Si el primer uno lógico aparece como +V, el segundo uno lógico aparecerá con el valor contrario es decir -V. El tercer uno del flujo de bits será +V y el cuarto -V, etc. Si se reciben dos bits consecutivos con un mismo nivel, +V o -V, esto quiere decir que alguno de los bits está equivocado. La frecuencia fundamental se presenta cuando se transmiten dos unos consecutivos. Como en el formato de señal sin retorno a cero, la máxima velocidad de transmisión es dos veces el ancho de banda del sistema.

2.10 Adecuación de la información

Existe un modo de transmisión conocido como banda base, en el cual los bits que conforman el mensaje se emiten por la línea bajo la forma de impulsos de corriente. Por ejemplo, se le asigna al uno lógico un valor positivo de tensión y al cero lógico un valor negativo de tensión (señal sin retorno a cero bipolar). El principal problema de éste modo de transmisión es que sólo es válido para longitudes cortas ya que las señales digitales se degradan rápidamente. Aunque, en principio, se podrían usar amplificadores para la transmisión de la señal en su forma digital, ésto no resulta ser práctico. Si se transmite una cadena consecutiva de ceros o unos, se generaría una señal de baja frecuencia, por lo que se necesitaría un medio de transmisión capaz de transmitirlas, y no debe olvidarse que el principal medio de transmisión de las redes, son las líneas telefónicas, que cortan las señales menores a 300 Hz.

Existen diferentes técnicas para evitar el problema de transmisión en banda base. Una de las soluciones más comunes es codificar los unos y ceros no como voltajes constantes sino como tonos de frecuencia. Por ejemplo un uno lógico podría ser transmitido como un tono de 1200 Hz y un cero como un tono de 1600 Hz. A este método de transmitir una señal usándola para variar alguna propiedad de otra señal, se le conoce como modulación. A la señal senoidal que se usa para transportar a la señal digital se le conoce como portadora. Los datos que modulan la portadora (los datos que proceden del terminal) constituyen la señal en banda base, es decir la señal no modulada.

Existen diferentes forma de modular una señal, se puede modular respecto a la amplitud, respecto a la frecuencia o respecto a la fase. La modulación en amplitud (Amplitude Switch Keying, ASK), consiste en modificar la amplitud de la señal portadora de acuerdo con el flujo de bits que se han de enviar. En este caso, una amplitud más elevada representa un cero, y una amplitud más baja representa un uno. Un sistema más extendido es la modulación en frecuencia (Frecuency Switch Keying, FSK), que consiste en variar la frecuencia, manteniendo constante la amplitud. Otro sistema es la modulación modulación en fase (Phase Switch Keying, PSK), que consiste en alterar de forma abrupta la fase para representar el cambio de uno a cero o de cero a uno.

2.11 Medios de transmisión.

Los medios de transmisión proveen los canales físicos para conectar los diferentes nodos de una red. Los medios de transmisión pueden ser divididos en guiados (clabe coaxual, par trenzado, fibra óptica, etc.) y no guiados (el aire, etc.) Aunque ya existen serios intentos de redes de área local inalámbricas, los medios guiados son los que más se utilizan en la construcción de redes de área local por lo que nos centraremos en ellos. La elección entre los diferentes medios de transmisión se deberá a las características de la red, su tamaño, sus tolerancia a ruido, la calidad de transmisión necesaria, etc.

2.11.1 Par trenzado

Uno de los medios más utilizados en la construcción de redes de área local es el par trenzado. El par trenzado consiste en un par de conductores de cobre cubiertos por una capa de aislante, que se trenzan uno alrededor del otro como se muestra en la figura 2.10. El calibre de los conductores oscila entre veintidós y veintiseis, el mismo que se utiliza en los sistemas telefónicos.

En las redes de área local el par trenzado se utiliza para aplicaciones de bajo costo y de bajo rendimiento. Este medio presenta las siguientes ventajas: es fácil de instalar, no requiere herramientas especiales para su instalación, por lo regular en todos los edificios ya existen sistemas de cableado en base a este medio y es más barato que los otros medios de transmisión. La distancia máxima que puede ser cubierta con este medio es de aproximadamente un kilometro. Para señales dentro del rango de frecuencias de la voz se pueden tener largos de hasta seis kilómetros sin necesidad de repetidoras.

El hecho de trenzar los cables uno alrededor del otro minimiza el efecto de la radiación externa. Si se le aplica un voltaje externo a uno de los cables, se aplica de igual manera al otro. De esta manera el trenzado elimina el efecto del ruido. Conforme aumenta el numero de gajos por unidad de longitud aumenta su poder de reducir las señales de ruido; desafortunadamente esto aumenta la cantidad de cable utilizado y por lo tanto aumenta su costo. La mayor parte de los cables trenzados utilizados en los sistemas telefónicos tienen de diez a quince gajos cada treinta centímetros.

Figura 2.10 Par Trenzado

El hecho de trenzar los cables uno alrededor del otro minimiza el efecto de la radiación externa. Si se le aplica un voltaje externo a uno de los cables, se aplica de igual manera al otro. De esta manera el trenzado realmente elimina el efecto del ruido externo. Conforme aumenta el numero de gajos por unidad de longitud aumenta su poder de reducir las señales de ruido; desafortunadamente esto aumenta la cantidad de cable utilizado y por lo tanto aumenta su costo. La mayor parte de los cables trenzados utilizados en los sistemas telefónicos tienen de diez a quince gajos cada treinta centímetros.

Existen dos tipos de cables de par trenzado, el par trenzado sin blindaje (Unshielded Twisted Pair UTP) y el par trenzado blindado (Shielded Twisted Pair STP). En el STP cada cable esta revestido mediante una malla que tiene como función proteger el cable de campos eléctricos externos. El UTP no tiene esta malla, no está blindado.

El UTP empieza a adquirir mayor importancia en el campo de las redes de área local por su gran oferta y por su costo reducido. En los primeros años de la década de los ochentas, existían pocas aplicaciones para el UTP por las altas velocidades de transmisión que se requieren para poder utilizar este medio. A partir de 1985 empiezan a parecer productos para redes de área local que soportan velocidades de uno a cuatro mega bits por segundo lo que hace posible el uso de UTP, aunque sea sólo en distancias cortas. Para 1990, aparecen en el mercado productos de hasta 16 megabits por segundo con UTP y están en discusión aplicaciones para velocidades de cien megabits por segundo.

Los cables UTP y STP para redes de tipo Ethernet y Token Ring deben cumplir con las siguientes especificaciones:

- Tener una impedancia entre 85 y 115 Ohms a 10 MHz.
- Presentar una atenuación máxima de 11 dB/110 mts. a 10 MHz o una atenuación máxima de 7.2 dB/110 mts. a 5 MHz.

2.11.2 Cable coaxial

Este medio debe su nombre a la forma de construcción del mismo, el cable se muestra en la figura 2.11. En el centro del cable se coloca un conductor, usualmente de cobre. El conductor se cubre por una capa aislante, que a su vez se reviste con una malla conductora que actúa como una armadura contra las señales de ruido externas. Como la armadura rodea completamente al conductor y tiene un eje común, la malla no sólo protege al conductor del ruido externo sino que previene que el cable genere campos eléctricos que a su vez puedan afectar a otros cables.

El cable coaxial ofrece un ancho de banda mayor que el brindado por el par trenzado, con la condición adicional de tener una alta inmunidad al ruido y una baja incidencia de errores. Su tamaño varía de acuerdo al ancho del conductor, de la malla y el aislante. Permite el uso de velocidades de diez megabits por segundo a lo largo de varios cientos de metros o inclusive de algunos miles de metros. Además es altamente inmune a interferencias electrónicas e interferencias de radio frecuencia. Entre más grueso sea el conductor del cable mayor será la distancia a la que puede ser transmitida una señal. El cable grueso suele ser más caro y menos flexible. Por tal razón, cuando tiene que colocarse en instalaciones en donde ya existen canales para cableado o conductos con espacio reducido y, sobre todo, limitado en las esquinas o dobleces, resulta más conveniente utilizar el cable delgado debido a que las nuevas instalaciones de ductos para cable por lo general son más costosas.

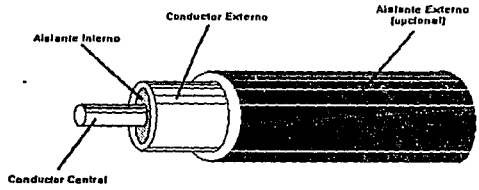


Figura 2.11 Cable Coaxial.

Fue el medio de transmisión más común en los años ochentas. Los cables de par trenzado, generalmente utilizados en aplicaciones telefónicas, no podían ser utilizados porque necesitaban velocidades de transmisión muy altas y fibra óptica se encontraba todavía en plan experimental, por lo que era muy cara. Este panorama cambio al principio de los noventas. Los sistemas electrónicos se mejoraron lo que permitió el uso de los cables de par trenzado y la fibra óptica se manufacturo en mayor cantidad con lo que se disminuyo su costo. Mientras el cable coaxial todavía es una opción importante dentro de las redes de área local, será utilizado muy poco en las redes de área metropolitana y menos aún en las de área amplia.

El amplio uso del cable coaxial, en otros sistemas de comunicación, ha hecho que tenga un costo moderado y que se fácil de conseguir en el mercado. Además las técnicas de instalación, conexión y control de transmisión, de este medio, se hayan bien desarrolladas. Existen una gran variedad de derivadores, divisores, acopladores, controladores, amplificadores para ser usados con este medio.

2.11.3 Fibra óptica

La naturaleza dieléctrica de la fibra óptica ha hecho de ella una muy buena alternativa como medio de transmisión. Sin lugar a dudas, la fibra óptica es el medio de transmisión por excelencia en las redes de área metropolitana y está empezando a ser ampliamente utilizado en las redes de área local. La fibra óptica es un delgado y flexible medio de transmisión que actúa como guía de señales comprendidas en un rango de 10^{14} a 10^{15} Hz, el cual incluye tanto a la luz visible como una parte del espectro infrarrojo.

Hace una década el entusiasmo por la fibra óptica era muy alto; se convirtió en el primer medio esencialmente inmune a cualquier tipo de interferencia eléctrica y capaz de transmitir información hasta velocidades de un billón de bits por segundo, con las únicas desventajas de ser caro y de utilizar dispositivos electrónicos también caros. Pero a finales de la década de los ochenta el costo de la fibra óptica empezó a disminuir, como consecuencia del perfeccionamiento de los métodos de producción de la misma, y poco a poco, el costo de los dispositivos electrónicos asociados a ella también empezó a disminuir. Estos dispositivos electrónicos son vitales para el uso de la fibra óptica como medio de transmisión, ya que lo que se transmite por medio de ésta no son señales eléctricas sino luz, por lo que se necesitan convertidores electro-ópticos. La señal eléctrica de entrada debe ser convertida a la señal óptica que será transmitida. Los generadores de la señal óptica más comunes son los diodos emisores de luz (Light Emmiting Diode LED) o el diodo láser (Injection Laser Diode ILD). Los LEDs son más baratos, pero sólo pueden ser utilizados en aplicaciones de baja velocidad.

La señal óptica se recibe mediante un fotodiodo que tiene la función de convertir los fotones que lo inciden en señales eléctricas. Los fotodiodos comúnmente utilizados son el fotodiodo positivo-intrínseco-negativo (Positive-Intrinsic-Negative PIN) y el fotodiodo de avalancha (Avalanche PhotoDiode APD). El PIN es más barato que el APD pero se usa sólo en aplicaciones de baja velocidad.

La fibra óptica es un medio de transmisión unidireccional para establecer un sistema bidireccional se requiere de un par de cables. Su estructura consiste en un solo cilindro dieléctrico sólido con un radio a y un índice de refracción n_1 (corazón de la fibra).

El corazón se encuentra rodeado por otro dieléctrico sólido, revestimiento, con un índice de refracción n_2 menor que n_1. Como el índice de refracción del corazón de la fibra es mayor que el del revestimiento, la luz tiende a propagarse a través de la fibra por medio de reflexión dentro del corazón del cable (ver figura 2.12). El corazón está generalmente construido de vidrio o de componentes plásticos, y tiene un diámetro de entre 2 y 125 micrones (μm). Adicionalmente las fibras se encuentran encapsuladas en un material elástico y resistente a la corrosión. El material exterior además de proteger a la fibra conductora del polvo y de otros agentes externos, le da cierta fuerza al cable.

Como se muestra en la figura 2.12, de acuerdo a la composición del corazón se tienen dos tipos de fibra, la de paso indexado y la de índice graduado. En el primer caso el índice de refracción es uniforme a través de todo el corazón, pero sufre un cambio abrupto (o paso) en el límite entre el corazón y el revestimiento. En las fibras de índice graduado el índice de refracción del corazón está en función de la distancia real desde el centro de la fibra.

Ambos tipos de líneas pueden ser divididas en fibras multimodo y de modo simple. Como su nombre lo indica las fibras de modo simple brindan sólo un modo de propagación, mientras que las multimodo soportan cientos de modos de propagación. En la figura 2.17 se muestran los tamaños típicos de fibras de modo simple y multimodo. Los tamaños más populares para fibras multimodo para redes de área local son: 50, 62.5, 85 y 100 μm de diámetro. El diámetro del corazón de las fibras de modo simple es de 9 μm con una longitud de onda de 1300 nm. Las fibras multimodo con índice graduado tienen un rango de ancho de banda distancia de algunos GHz-Km, mientras que las modo simple tienen capacidades que exceden esto.

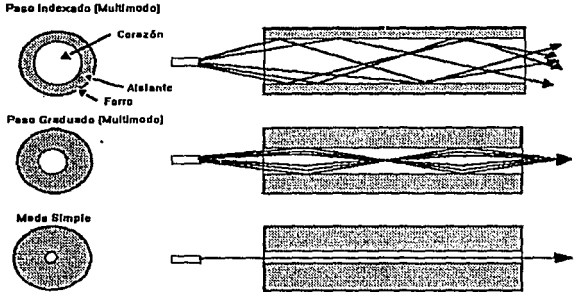


figura 2.12 Características de Transmisión de una fibra multimodo y una de modo simple.

Capitulo III

Normas y Arquitectura de Redes

Los primeros diseños de redes se caracterizaron por ser sistemas propietarios, como cada uno de los diferentes fabricantes de sistemas de cómputo usaba las técnicas que le parecian más adecuadas para la creación de sus sistemas, era muy difícil conectar sistemas de fabricantes distintos. Cada fabricante tenía la libertad de establecer su propio conjunto de reglas para efectuar enlaces y/o transmisión de datos entre los equipos que formaran la red. Entre las primeras compañías en tratar de realizar sistemas de red se encontraban IBM, General Motors y AT&T quienes trataron imponer sus protocolos como estándar en base a su posición predominante en el mercado. Sin embargo todos los días aparecian nuevas compañías fabricantes de sistemas que no se ajustaban al estándar por lo que los problemas de conectividad seguían existiendo. El caos generado por la incompatibilidad entre los diferentes sistemas dio lugar a la exigencia de los usuarios, para que se estableciera una normalización al respecto.

Esta normalización no solamente iba a facilitar la comunicación entre ordenadores construidos por diferentes compañías, sino también traería, como beneficio, el incremento en el mercado para los productos que se plegaran a norma establecida, lo que conduciria a una producción masiva, una economía de escala por incremento de la producción, nuevos desarrollos aprovechando las nuevas técnicas, así como otros tipos de beneficios cuya tendencia sería disminuir el precio de los productos y alentar su posterior aceptación.

Las normas se dividen en dos categorías que pueden definirse como: de facto y de jure. Las normas *De Facto* (derivado del latín, que significa "del hecho"), son aquellas que se han establecido sin ningún planeamiento formal. Las normas IBM PC y sus sucesoras son normas de facto para ordenadores. Similarmente UNIX es la norma de facto para los sistemas operativos de los departamentos de ciencias de la computación en las universidades.

En contraste, las *De Jure* (derivado del latín, que significa "por ley") son normas formales, legales, adoptadas por un organismo que se encarga de su normalización. Existen diferentes organizaciones encargadas de establecer los estándares, las principales organizaciones de estándares, como es de suponerse son estadounidenses o europeas. El objetivo de estas organizaciones es el de fungir como arbitro entre los diferentes fabricantes para negociar el establecimiento de un estándar, sin embargo en ocasiones ni siquiera las diferentes organizaciones pueden ponerse de acuerdo en cual será el estándar y mientras una de estas organizaciones acepta una técnica como estándar otra organización establece una diferente. Como no existe forma legal para obligar a los fabricantes a usar el estándar, ya que ni siquiera existe un estándar único, los estándares se establecen como una sugerencia, como una recomendación y no como una orden que debe ser cumplida.

3.1 Normas para redes

Las normas de red de área local comúnmente aceptadas son: el modelo de referencia para la interconexión de sistemas abiertos o modelo OSI (Open System Interconection) establecido por la Organización Internacional de Estándares (ISO) y la norma 802 del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). La arquitectura FDDI es otra de las normas de interés. Esta norma fue desarrollada por la ANSI, para la construcción de redes en base a fibra óptica. A continuación se da una explicación del modelo de referencia OSI, de las normas IEEE 802 y de la norma para fibra óptica FDDI.

3.1.1 Modelo de Referencia OSI

El modelo de referencia OSI es un modelo que se organiza en siete de capas o niveles, cada una de las capas se construye sobre su predecesora. El propósito de cada capa es ofrecer ciertos servicios a las capas superiores, sin que estas tengan que interesarse en el conocimiento detallado de cómo se realizan dichos servicios. La capa *n* en una máquina conversa con la capa *n* de otra máquina. Las reglas y convenciones utilizadas en esta conversación se conocen conjuntamente como protocolo de la capa *n*. En realidad no existe una transferencia directa de datos desde la capa *n* de una máquina a la capa *n* de otra; por el contrario, cada capa pasa información de datos y control a la capa inmediatamente inferior, y así sucesivamente hasta que se alcanza la capa localizada en la parte más baja de la estructura.

La comunicación entre la capa inferior de cada máquina permite que la información viaje de una máquina a la otra. Una vez que el mensaje se encuentra en la máquina receptora el proceso se invierte y la información asciende hasta la capa *n* de la máquina destino.

Entre cada par de capas adyacentes hay una interfase, la cual define los servicios y operaciones que la capa inferior ofrece a la superior.

Al conjunto de capas y protocolos se le denomina arquitectura de red. Aquí cabe aclarar que el modelo OSI, por sí mismo, no es una arquitectura de red, dado que no especifica, de forma exacta, los servicios y protocolos que se utilizarán en cada una de las capas. Sólo indica lo que cada capa deberá hacer.

En la figura 3.1 se muestran los siete niveles del modelo OSI. Cada una de estas capas (o niveles) del modelo tienen sus propias funciones las cuales serán descritas a continuación.

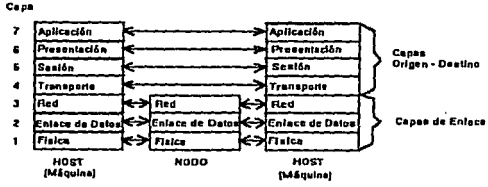


Figura 3.1

Modelo de Referencia OSI

Capa 1 (Aspecto físico).- La capa física se encarga de la transmisión de bits a lo largo de un medio de comunicación. Define las características físicas del medio de transmisión, establece sus especificaciones eléctricas y mecánicas, es decir controla el intercambio de información a nivel de bits sobre el medio de transmisión. Además, controla lo relacionado con la velocidad de transmisión, codificación de la información y conexión física de los diferentes dispositivos. Esta capa además de tratar la transmisión de bits entre entidades físicas, ofrece los mecanismos necesarios para la activación, mantenimiento y desactivación de las conexiones a nivel físico. Adicionalmente es capaz de determina si la codificación será numérica/analógica, en modo síncrono o asíncrono, etc.

Capa 2 (Enlace de datos).- La función de esta capa es controlar tanto el enlace lógico, como la detección y corrección de errores. Ofrece la entrega confiable de datos a través del enlace físico. Esta capa es la responsable de la validez e integridad de la transmisión de nodo a nodo, cubriendo adicionalmente la sincronización de la transmisión. Su tarea primordial consiste en, a partir de un medio de transmisión común y corriente, construir una línea de transmisión sin errores para la capa de red (capa inmediata superior). Esta tarea la realiza al hacer que el emisor troce la entrada de datos en unidades de información, llamados paquetes (típicamente constituidos por algunos cientos de octetos), obliga al transmisor a enviar los paquetes en forma secuencial y a procesar los paquetes de asentimiento, devueltos por el receptor. Como la *capa 1*, básicamente acepta y transmite un flujo de bits sin tener en cuenta su significado o estructura, recae sobre la capa de enlace la creación o reconocimiento de los límites del paquete.

El paquete puede destruirse por completo debido a una ráfaga de ruido en la línea, en cuyo caso el software de la capa de enlace, perteneciente a la máquina emisora, deberá retransmitir el paquete. Como múltiples transmisiones del mismo paquete, podrían producir la duplicación del mismo corresponde a esta capa resolver ta,bién los problemas causados duplicidad de los paquetes.

Capa 3 (Capa de Red).- Esta capa se encarga del control de la red, coordina la conexión entre nodos adyacentes y se encarga del ruteo y manejo de los paquetes. Administra conexiones a través de la red para las capas superiores. Un punto de suma importancia en su diseño, es la determinación sobre cómo encaminar los paquetes del origen al destino. Las rutas podrían basarse en tablas estáticas que se encuentran "cableadas" en la red y que difícilmente podrían cambiarse. También, podrían determinarse al inicio de cada conversación, por ejemplo en una sesión de terminal. Por último, podrían ser de tipo dinámico, determinándose en forma diferente para cada paquete, reflejando la carga real de la red. Controla el número de paquetes en la red: para evitar congestionamientos, o para saber cuantos paquetes se enviaron a cada cliente para su eventual facturación.

Capa 4 (Capa de Transporte).- Como su nombre lo dice se encarga de controlar el transporte de la información. Su función principal consiste en aceptar los datos de la capa de sesión (capa superior), dividirlos, siempre que sea necesario, en unidades más pequeñas (paquetes), pasarlos a la capa de red (capa inferior) y asegurar que todos ellos lleguen correctamente al otro extremo. Bajo condiciones normales, la capa de transporte crea una conexión de red distinta para cada conexión de transporte solicitada por la capa de sesión.

La capa de transporte maneja ruteos entre nodos múltiples y rutas alternas, ofrece detección y corrección de errores de punta a punta (end-to-end) , por medio del cual se entregan los mensajes en el mismo orden en que fueron enviados (servicio orientado a conexión); o bien puede encargarse del transporte de mensajes aislados sin garantizar el orden de distribución (servicio no orientado a conexión).

Capa 5 (Capa de Sesión).- La capa de sesión permite que los usuarios de diferentes máquinas puedan establecer sesiones entre ellos. Una sesión podría permitir al usuario acceder a un sistema de tiempo compartido a distancia, o transferir un archivo entre dos máquinas. Uno de los servicios de la capa de sesión consiste en gestionar el control de diálogo. Las sesiones permiten que el tráfico vaya en ambas direcciones al mismo tiempo, o bien, en una sola dirección en un instante dado. Si el tráfico sólo puede ir en una dirección en un cierto momento, la capa de sesión ayudará en el seguimiento de quien tiene el turno.

Capa 6 (Capa de Presentación).- La capa de presentación realiza funciones de uso frecuente para las cuales es más cómodo establecer una solución general. A diferencia de las capas inferiores, que únicamente se ocupan del movimiento fiable de bits de un lugar a otro, la capa de presentación se ocupa de los aspectos de sintaxis y semántica de la información que se transmite. Los servicios brindados por la capa de presentación son: conversión de códigos, establecimiento de terminales virtuales y trasferencia de archivos. Estándariza la presentación de datos a las aplicaciones. Su tarea es controlar la forma como los datos son representados, de tal manera que puedan ser intercambiados entre sistemas con diferentes representaciones internas.

La capa de presentación está relacionada también con otros aspectos de representación de la información. Por ejemplo, la compresión de datos que tiene por objetivo reducir el número de bits a ser transmitidos o el proceso de criptografía utilizado para establecer mecanismos de privacidad, son procesos relacionados con la capa de presentación.

Capa 7 (Capa de Aplicación).- La capa de aplicación interactúa con las aplicaciones de los usuarios y con los sistemas operativos locales. Se forma, como su nombre claramente lo establece, con los programas de aplicación que se utilizan en la red. Maneja los recursos para las aplicaciones de red como correo electrónico, transferencia de archivos, terminales virtuales, bases de datos, etc. Ofrece protocolos para diversos servicios y aplicaciones, su principal propósito es la comunicación entre programas. Esta capa define las reglas para entrar en el sistema de comunicación, los programas se comunican unos con otros a través de ella.

3.1.2 Estándar 802 del IEEE

Este estándar se forma principalmente por tres propuestas, cada una de ellas forma una instancia del estándar:

Norma (IEEE 802.3).- El estándar IEEE 802.3 describe las características de una red de área local con topología de bus y CSMA/CD como controlador de acceso al medio (MAC). Este estándar tiene como antecedentes los esfuerzos realizados por empresas independientes, para integrar un procedimiento que permitiera la comunicación entre computadoras en la red conocida como Ethernet.

- El equipo físico Ethernet: El equipo físico lo forman las tarjetas o controladores, puentes y cables. Cada tarjeta Ethernet tiene una dirección otorgada por el fabricante. La dirección Ethernet tiene un identificador único de 48 bits. Los dos primeros bits son una bandera; el primero indica si la dirección es local o administrada universalmente.

- Los paquetes Ethernet: Los paquetes son un conjunto de bits que se transmiten por el equipo físico Ethernet. Entre los elementos importantes del paquete Ethernet tenemos el domicilio del remitente, el domicilio del destinatario, y el campo de datos. Los paquetes Ethernet son formados y transmitidos por el controlador. Este controlador también tiene que ser capaz de reconocer los paquetes que son dirigidos a él.

Aunque originalmente la norma 802.3 fue pensada para utilizar cable coaxial como medio de transmisión, en la actualidad el estándar puede utilizar, además de cable coaxial, fibra óptica y par trenzado sin blindaje (UTP) como capa física.

El control de acceso a los medios se efectúa por medio de un marco de bits que contienen la información necesaria para llevar a cabo dicho control. La figura 3.2muestra el marco utilizado por el estándar 802.3 para el control de acceso al medio. Los campos del marco y su significados son los siguientes:

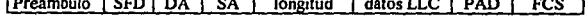


Figura 3.2Formato del marco del estándar 802.3

Preámbulo: Usado para la sincronización del reloj. Consiste en el siguiente patrón de bits: "10101010". El preámbulo tiene una longitud de 7 octetos.

Inicio del marco (Start Frame Delimiter SFD): denota el inicio del marco, el patrón SFD es "10101011". Tiene un sólo octeto de longitud.

Dirección destino (Destination Address DA): El controlador de acceso al medio (MAC) necesita marcar la dirección de la estación que recibirá el marco. El tamaño de esta parte del marco puede ser de 2 o de 6 octetos.

Dirección origen (Source Address SA): Campo donde el MAC señala la dirección de la estación que envía el marco. Como en la anterior este campo pude ser de 2 o de 6 octetos.

Longitud: Señala el número de octetos que se encuentran el campo siguiente, datos LLC. Este campo tiene un tamaño de 2 octetos.

Datos del enlace lógico (Logic Link Control LLC): Datos del control de enlace lógico y capas superiores. El tamaño de este campo puede ir desde 0 hasta 1500 octetos.

PAD: Octetos adicionales para asegurarse que el marco es de al menos 64 octetos. La longitud mínima para el PAD y el campo de datos LLC es de 46 octetos.

Marco de revisión de secuencia (Frame Check Sequence FCS): campo donde se guarda información útil para detección de errores.

El marco es seguido por un tiempo de 96 bits (9.6 μs a una velocidad de 10 Mbps) de silencio en la línea para marcar el fin del marco.

Norma (IEEE 802.4): La norma 802.4 físicamente es un cable lineal, o en forma de árbol, al cual se conectan las estaciones, pero donde las estaciones están lógicamente organizadas en un anillo, en el que cada una de las estaciones conoce la dirección de la estación ubicada a su "izquierda" y "derecha".

Cuando el anillo lógico se inicia, la estación que tiene el número mayor es la que puede iniciar la transmisión. Después de que ésta lo hizo, pasa el token a su vecino inmediato, para que éste transmitir información. Como solamente una estación puede tener el token a la vez, no hay posibilidad de colisiones. Recuerde el ejemplo, planteado en el capítulo uno, de la mesa de discusión en la que sólo la persona que tiene el micrófono puede hablar.

Un punto interesante es que el orden físico en el que se encuentren conectadas las estaciones al cable no es importante. Aunque todas las estaciones pueden "ver" todos los mensajes, sólo recibien los paquetes enviados a ellas y descartar los que no lo estén. Cuando una estación pasa el token, lo pasa específicamente a su vecino lógico en el anillo, independientemente del lugar físico en donde se encuentre la estación en el cable. (Ver figura 1.11). Como la estrategia de token es relativamente más complicada de construirr en una topología de bus que en una de anillo, y como la técnica de contención (CSMA/CD) es más fácil de implementar en un bus, los productos basados en el estándar 802.4 por lo general son más caros que otros productos para redes de área local.

Para la capa física, las redes token bus utilizan cable coaxial de banda ancha de 75 ohms, que normalmente se emplea para la televisión por cable. Tanto el sistema de cable sencillo como dual están autorizados, con o sin repetidores centrales.

Como el estándar anterior la norma 802.4 utiliza un grupo de bits para controlar el acceso al medio, los campos que forman el marco de control pueden verse en la figura 3.3 y son explicados a continuación.

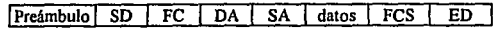


Figura 3.3 Formato del marco del estándar 802.4

Preámbulo: Usado para la sincronización de los bits. Por lo general tiene una longitud de uno o dos octetos. Debe tener una duración de al menos 2μs.

Inicio del marco (Start Delimiter SD): Indica el primer octeto del marco. Tiene sólo un octeto de longitud.

Control del marco (Frame control FC): Indica si el marco es un marco de datos de enlace lógico (Logic Link Control - LLC data), si es un marco de control de acceso al medio (MAC) o si se trata de un marco para el control de la estación. Este campo tiene una longitud de un octeto.

Dirección destino (Destination Address DA): El controlador de acceso al medio (MAC) necesita marcar la dirección de la estación que recibirá el marco. El tamaño de esta parte del marco puede ser de 2 o de 6 octetos.

Dirección origen (Source Address SA): Campo donde el MAC señala la dirección de la estación que envía el marco. Como en la anterior este campo pude ser de 2 o de 6 octetos.

Datos: Esta campo contiene un marco de enlace lógico de datos (LLC data frame) o un datos para el control de acceso al medio. Debe tener una longitud mayor o igual a 8182 octetos si tiene 2 octetos de dirección y debe tener 8174 octetos si la dirección es de 6 octetos.

Marco de revisión de secuencia (Frame Check Sequence FCS): campo donde se guarda información útil para detección de errores.

Delimitador del marco (End Delimiter ED): Indica el fin del marco, tiene una longitud de un octeto.

Norma (IEEE 802.5): Esta norma establece un protocolo conocido con el nombre de "Token Ring". El nombre Token Ring se le atribuye debido a su forma de acceso por medio del token y por su topología en anillo (Ring), aunque es común encontrar redes token ring implementadas en topologías estrella. Cada nodo recibe información de uno de sus vecinos, el mas cercano según la corriente ascendente, (Nearest Active Upstream Neighbor NAUN) transmite la información al nodo inmediato descendente. En una red Token Ring, el token tiene una longitud de 24 bits.

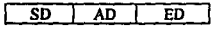
Una red Token Ring tiene la posibilidad de dar prioridades de acceso, y desconectar nodos descompuestos entre otras características. En la tecnología Token Ring es muy común tener lo que se conoce como puente (bridge). En términos generales un puente es un nodo que se encuentra al mismo tiempo en dos anillos y es capaz de pasar información de un anillo a otro.

Existe un tipo especial de puente, el conocido como: conector o dorsal de anillos (backbone ring), el cual conecta varios anillos entre si. Este conector especial, consiste en una serie de puentes, cada uno conectado a una red local.

Cada nodo en la red utiliza un sólo cable con dos pares para conectarse a un dispositivo llamado la unidad de acceso al medio (MAU). Un par se dedica a recibir datos y otro par para enviar. Cada nodo se conecta al controlador vía un conector de 4 vías.

Los datos en una red Token Ring se transmiten a velocidades de 4 Megabits por segundo y también a 16 Mbps. Se pueden conectar hasta 255 nodos. Los campos del marco de control y del token se muestran en la figura 3.4.

Token:



Marco:

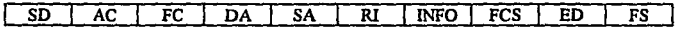


Figura 3.4 Formato del marco del estándar 802.5

Inicio del marco (Start Delimiter SD): Marca el inicio de la transmisión. Se forma por el patrón JK0JK00. El tamaño de este campo es de un octeto.

Control de Acceso (Access Control AC): Indica si el mensaje es un marco o un token y contiene información acerca de la prioridad de la información.

Control del marco (Frame control FC): Indica si el marco contiene datos de enlace lógico (Logic Link Control - LLC data) o si contiene información para el control de acceso al medio (MAC). Este campo tiene una longitud de un octeto.

Dirección destino (Destination Address DA): Dirección de la estación que recibirá el marco. El tamaño de esta parte del marco puede ser de 2 o de 6 octetos.

Dirección origen (Source Address SA): Campo donde el MAC señala la dirección de la estación que envía el marco. Como en la anterior este campo pude ser de 2 o de 6 octetos.

Información de ruteo (Routing Information RI): Campo opcional, usado para redes múltiples que usan la estrategia de "ruteo de origen", en las cuales el destino se encuentra en un anillo diferente a donde se encuentra el origen. En el "ruteo de origen" el transmisor puede especificar el camino que debe seguir la información para llegar a su destino. El tamaño de esta campo oscila entre cero y dieciocho octetos.

Información (Information INFO): Contiene un marco LLC o información para el MAC. El estándar no especifica una longitud máxima para este campo, pero su tamaño estará limitado por el tiempo requerido para transmitir el marco.

Marco de revisión de secuencia (Frame Check Sequence FCS): campo donde se guarda información útil para detección de errores.

Delimitador del marco (End Delimiter ED): Indica el fin del marco, se forma por el patrón de bits JK1JK1IE. Este campo también indica si el marco es el último de una secuencia de múltiples marcos (I) y si se detectó un error en el receptor (E). Este campo es de un octeto.

Estado del marco (Frame Status FS): Patrón de bits AC00AC00; estos bits indican si la dirección destino del marco fue reconocida por cualquier estación en la red de donde se emitió y si fue copiada exitosamente por alguna estación, en una red externa a la red de transmisión.

3.1.3 Interfase de Datos Distribuidos por Fibra Óptica (FDDI)

La Interfase de Datos Distribuidos por Fibra Óptica (Fiber Distributed Data Interface FDDI), es un estándar de la ANSI, para redes de alta velocidad, recomendado para la construcción de la columna vertebral (backbone) de redes de propósito general, que utilicen fibra óptica como medio de transmisión. En general este estándar se recomienda para redes de área local de alta velocidad (HSLN) o para redes de área metropolitana (MAN). Permite la interconexión de más de 500 dispositivos, trabajando a una velocidad de 100 Mbps sobre distancias mayores de 100 km. Este apartado describirá los conceptos generales de está arquitectura.

El conjunto de estándares para FDDI fue desarrollado con el objetivo de brindar una interconexión de propósito general y de alta velocidad, entre computadoras de alta velocidad y sus periféricos, incluyendo la interconexión entre redes. De esta forma una red FDDI puede actuar como una HSLN para interconectar un gran número de sistemas de cómputo dentro de una área reducida o puede actuar como una red de área amplia conectando redes pequeñas a través de una gran área. Puede adicionalmente, usarse para conectar redes locales con redes de área amplia. Sus características son:

- Se basa en el estándar IEEE 802.5, por lo que usa Token passing como esquema de control de acceso al medio.

- Compatibilidad con las redes de área local basadas en el IEEE 802 mediante el control lógico para el enlace de datos.

- Tiene la habilidad para usar fibra óptica multimodo o modo simple e inclusive par trenzado, para su construcción.

- Usa la topología de doble anillo, para poder soportar fallas en alguno de sus nodos.

- Opera a una velocidad de 100 Mbps y tiene la habilidad de sustentar una velocidad de transmisión efectiva de 80 Mbps.

- La posibilidad "teórica" de poder conectar cualquier cantidad de máquinas, aunque el estándar asume que no se pueden conectar más de 1000 unidades.

- La habilidad de colocar el ancho de banda de forma dinámica, lo que permite la transmisión tanto sincróna como asincróna simultáneamente.

En el estándar se han descrito cuatro ambientes de aplicación para las redes construidas a base de FDDI. Estos ambientes difieren principalmente por el número de estaciones conectadas y por el tamaño geográfico de la red, como se muestra en la figura 3.5. En el área del "centro de datos", actúa como una HLSN. Esta región se caracteriza por conectar un pequeño número de estaciones (no más de 50), la mayoría de estas estaciones son mainframes o computadoras veloces y dispositivos periféricos. La confiabilidad, alta velocidad, y tolerancia a las fallas son sus características principales, muchas de las estaciones tienen doble conexión para soportar fallas. En este ambiente se asume que la longitud de la fibra no debe ser mayor de 400 m entre las máquinas adyacentes y que el tamaño total del anillo no debe exceder 20 km.

En el edificio de oficinas, parte frontal de la red, se caracteriza por: un número relativamente alto de máquinas, no soportar fallas (las estaciones tienen una sola de conexión) y por el uso de la topología estrella. Las estaciones en este ambiente de aplicación son típicamente mini-computadoras, concentradores, estaciones de trabajo, computadoras personales o equipos periféricos.

En la región llamada campus, o columna (backbone) de la red, se caracteriza por las estaciones distribuidas a través de múltiples áreas, conectados mediante enlaces punto a punto en longitudes no mayores a 2 km. Este ambiente de aplicación se usa de forma típica como líneas de transmisión entre redes localizadas en diferentes edificios. Podemos decir que de esta forma las redes FDDI pueden ser usadas para la construcción de redes privadas.

Finalmente, en el ambiente multi-campo se caracteriza por grupos de estaciones localizadas en diferentes sitios, posiblemente separados por distancias de más de 60 km. En este ambiente es posible tener que usar vías de comunicación pertenecientes u operadas por alguna agencia, ya sea gubernamental o privada. El ambiente múlticampo es un ejemplo como las redes pueden ser utilizadas como redes públicas.

En muchos textos se considera a cada una de las máquinas (hosts) conectadas a la red como nodos, sin embargo en el contexto del estándar FDDI estos son dos conceptos diferentes. Para FDDI un nodo es un elemento activo en la red, un elemento que es capaz de retransmitir la información que recibe, aunque no sea capaz de corregir erroresen la misma. Por el contrario, una estación (host) FDDI es un nodo con una dirección en la red, capaz de transmitir, recibir y generar información. De acuerdo a este estándar todas las estaciones son nodos, pero no todos los nodos son estaciones.

Para terminar con la discusión del estándar FDDI sólo debemos aclarar el término de concentrador. Un concentrador es cualquier nodo en la red que además del puerto por medio del cual se conecta a la red cuenta con puertos adicionales. El concentrador puede ser utilizado como un multiplexor del tráfico de la red.

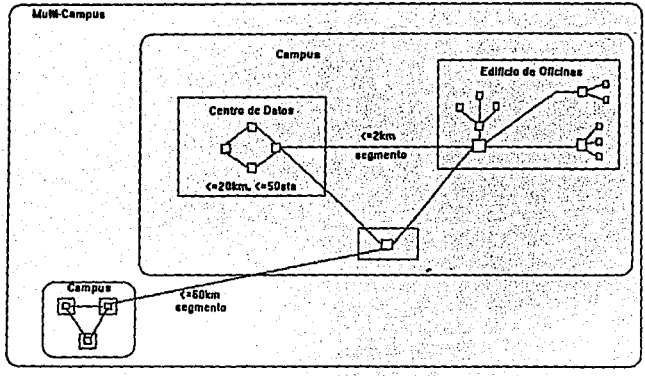


figura 3.5 Ambientes de aplicación de FDDI

3.2 Interconexión de redes

Los conceptos que hasta ahora se han presentado se establecieron pensando que sólo hay una red homogénea., con cada una de las máquinas utilizando el mismo protocolo en cada capa. Esto fue hecho para facilitar la comprensión de los conceptos hasta aquí expresados; sin embargo, el título de nuestra tesis marca que estudiaremos *sistemas abiertos*. El interés en el análisis de este tipo de sistemas se debe a que la mayoría de los sistemas de red utilizados en la actualidad son sistemas abiertos, además un punto de interés en el estudio de las redes es su aplicación en sistemas de gran escala, la RedUNAM o la red internacional "Internet", por ejemplo.

Es claro que sistemas de estas dimensiones no pueden estar formados por sistemas propietarios. En este apartado nos enfocaremos, en los componentes que nos permiten la interconexión de varias redes de área local para formar redes más grandes. La interconexión entre redes se da entre todos los tipos de redes, LANs, MANs y WANs. La figura 3.6 muestra las posibles combinaciones.

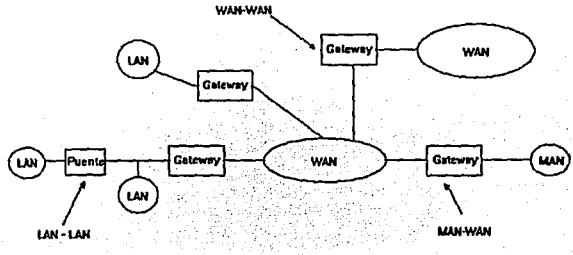


figura 3.6 Redes interconectadas

El tipo de interconexión más común es entre dos redes de área local. Esto se debe sobre todo a que este tipo de redes tiene un límite en el número de estaciones. Por ejemplo, las redes Token Ring están limitadas a 70 estaciones, distribuidas en un radio de unos cuantos kilómetros, la solución a esto es la construcción de múltiples redes de área local interconectadas de alguna manera.

Existen cuatro dispositivos diferentes para la interconexión de redes[1]: repetidores (repiters), puentes (bridges), ruteadores (routers) y pasarelas (gateways). Esta terminología es, en cierta forma, estándar; pero, no universal. La relación entre estos dispositivos con el modelo OSI y los protocolos relacionados se muestra en la figura 3.7.

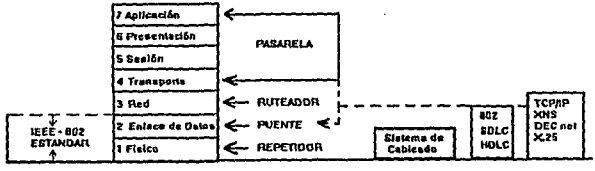


Figura 3.7 Relación de repetidores, puentes, ruteadores y pasarelas para el Modelo de Referencia OSI.

3.2.1 Repetidores (Repiters)

Como se mencionó en el capitulo dos, un repetidor tiene la función de recibir una cadena de bits y retransmitirla. Un sistema puede estar constituido por varios segmentos de cable y varios repetidores, pero no es posible que más de dos transmisores-receptores se encuentren separados por una distancia mayor de 2.5 km, ni tampoco es posible que exista una trayectoria entre dos transmisores-receptores, que atraviese más de cuatro repetidores.

[1] Algunos autores consideran a los cuatro dispositivos como retransmisores, sólo los diferencian de acuerdo a las funciones adicionales que realizan.

3.2.2 Puentes (Bridges)

Un puente (bridge), es el dispositivo encargado de conectar dos redes locales, su función es direccionar mensajes entre redes. De acuerdo a la dirección del mensaje, lo envía a través de una red o la otra, por lo general se utilizan para interconectar redes similares. La interconexión entre las redes puede ser punto a punto, es decir el puente de una red conectado al puente de la otra red; o bien puede que las dos redes están conectadas a un mismo puente En la figura 3.8 se muestra un diagrama de dos redes conectadas, de forma punto a punto.

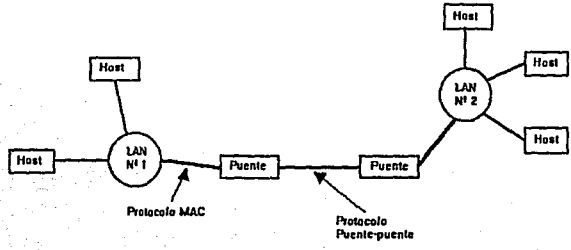


Figura 3.8 Puentes en un ambiente de Redes de Área Local.

A diferencia de los repetidores ordinarios que únicamente se dedican a pasar bits a través de ellos, los puentes examinan cada paquete y sólo re-expiden aquéllos que necesitan pasar de una red a otra.

De forma operacional, los puentes se consideran como otra estación conectada a la red. Como todas las estaciones el puente examina la dirección destino del mensaje, si la dirección del mensaje, indica que se trata de un envío de una estación de la "LAN 1", hacia una estación de la "LAN 2", copia el mensaje para después retransmitirlo hacia otro puente (conexión punto a punto) o a través de la LAN 2. La conexión punto a punto permite que el protocolo entre puentes sea diferente al protocolo utilizado por las redes. Esto resulta atractivo porque la conexión punto a punto no requiere de sistemas de control muy complicados, como los utilizados por un sistema de transmisión por canal (broadcast).

Los puentes son dispositivos que operan con la filosofía de almacenar y seguir y se utilizan para conectar redes que utilicen el mismo sistema de direccionamiento para llevar a cabo el control de acceso al medio, siempre y cuando ambas redes usen el mismo protocolo; operan en la capa física (capa 1) y en la capa de conexión de datos (capa 2).

3.2.3 Ruteadores (Routers)

Los sistemas grandes se dividen en subredes[2], la división de la red nos permite restringir el acceso algún subsistema, podemos hacer que cierto grupo de usuarios pueda accesar a partes del sistema mayor, pero no a todo el sistema, etc. Para poder realizar esta subdivisión se utilizan los ruteadores. La figura 3.9 muestra un sistema que utiliza tres ruteadores, para conectar tres redes diferentes.

[2] El término de sub-red no significa subordinación, sólo indica que diferentes redes se unen para formar una red mayor. Todas las su-bredes pueden operar de forma independiente, su relación sólo tiene por objetivo formar un sistema con mayores funciones.

Un ruteador es un dispositivo capaz de seleccionar un recorrido óptimo para comunicar una estación con otra y encaminar un mensaje de acuerdo a esta trayectoria. El ruteador opera en la capa de red del modelo OSI (capa 3). Los ruteadores se utilizan para conectar redes que trabajan con un mismo protocolo, se emplean sobre todo en redes donde se tienen múltiples vías de comunicación entre los usuarios. Un ruteador examina tanto la dirección de origen como la del destino del mensaje y determina la ruta mas efectiva. Los ruteadores son similares a los puentes, pero mientras los segundos se utilizan para conectar redes que utilizan el mismo sistema de direccionamiento para el control de acceso al medio, los ruteadores pueden conectar redes con diferentes sistemas de direccionamiento.

Otra diferencia entre un puente y un ruteador es, que mientras los puentes brindan una conexión punto a punto entre redes los ruteadores actúan como un conmutador (switch) entre varias redes.

Los ruteadores permiten que las redes difieran en sus características físicas, por ejemplo, se puede unir una red que utilice cable coaxial con una que utilice par trenzado, también puede dirigir mensajes a través redes con diferente topología pero con el mismo protocolo. Como opera en la capa de red del modelo OSI, la dirección de los paquetes puede monitorearse y utilizarse para administrar la red. Los ruteadores se utilizan para cualquier tipo de redes sin importar su extensión; LAN, MAN o WAN.

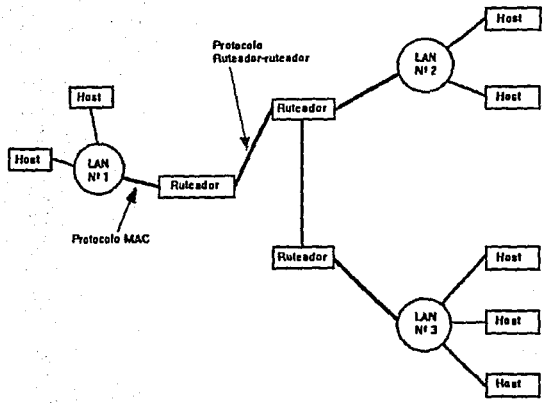


Figura 3.9 Ruteadores en un ambiente de Redes de Área Local.

3.2.4 Pasarelas (Gateway)

Las pasarelas (gateway) se utilizan para conectar redes heterogéneas, como se muestra en la figura 3.10. Las pasarelas son generalmente dispositivos muy inteligentes, que trabajan en el nivel siete del modelo OSI, es decir en la capa de aplicación, lo que les permite efectuar tanto funciones de direccionamiento como conversión de protocolos. Las pasarelas se utilizan para conectar redes de área local con otro tipo de redes, por lo general con redes de área amplia, esto se debe a su capacidad de convertir protocolos.

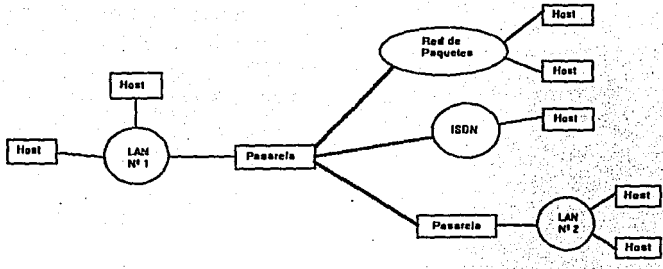


Figura 3.10 Pasarelas.

La figura 3.10 muestra un ejemplo de un sistema que utiliza pasarelas. Existen dos tipos de pasarelas, las orientadas a conexión y las no orientadas a conexión.

3.3 Protocolos

Todas las redes de comunicaciones están basadas en protocolos o reglas. Estas reglas definen cómo se prepara un mensaje; cómo se establece un canal de comunicaciones y cómo se controla la comunicación una vez establecida. Generalmente, los protocolos son normas públicas definidas por comités. En una situación ideal, sólo debería existir un conjunto de protocolos, y todo sistema de computadoras debería poder comunicarse con cualquier otro, pero debido a los intereses que se hayan detrás del establecimiento del mismo. En la actualidad existen muchos protocolos, cada uno utilizado como estándar dentro de su campo de aplicación.

Existen muchos protocolos de redes, a saber: TCP/IP, IPX, Netbios, Netbeui, etc. Cada uno de estos protocolos, tiene diferentes características, cuidando los aspectos para los cuales fue creado. El tipo de red determinará el protocolo a utilizarse, por ejemplo las redes Novell manejan IPX, las redes IBM utilizan Netbios, las redes Microsoft prefieren Netbeui y las redes en base a UNIX por lo general utilizan TCP/IP. La red "Cuautitlán 2", por ser una red UNIX usa este último como protocolo de comunicación.

El protocolo TCP/IP se forma por la unión de dos protocolos el protocolo de control de transmisión (TCP) y el protocolo de interconexión de redes (IP). Estos dos protocolos se estudiarán de forma separada para facilitar su comprensión.

En la figura 3.11 se muestra la arquitectura del protocolo TCP y algunos de los protocolos superiores relacionados con este. Las diferentes opciones en los protocolos superiores dependerán de las necesidades del usuario. En este trabajo nos encargaremos además del estudio de los protocolos TCP e IP de dos de los protocolos de nivel superior que se encuentran como estándar en las máquinas conectadas a Internet: Telnet y FTP.

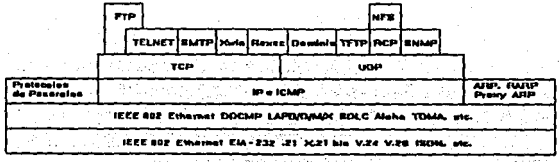


Figura 3.11 Estructura de los protocolos de red

3.3.1 Protocolo de interconexión de redes (IP).

El protocolo de interconexión de redes (internet Protocol IP)[3] fue desarrollado como parte de la red a cargo de la Agencia de Proyectos Avanzados de Investigación de Defensa DARPA (Defense Advanced Project Research Agency DARPA). La importancia de este protocolo lo han convertido en un estándar, de hecho muchos de los conceptos establecidos por el estándar ISO 8473 se derivaron del protocolo IP.

El protocolo IP ofrece de comunicación no orientado a conexión (connectioless), por lo tanto permite el intercambio de información entre máquinas sin la necesidad de establecer un proceso para iniciar la conexión; los problemas de establecimiento de comunicación, etc; sonresueltos por el protocolo TCP. Por ser, el IP, un protocolo no orientado a conexión, cuando se usa no se puede estar seguro que todos los paquetes llegarán a su destino. Por ejemplo una pasarela (gateway) IP tiene una longitud máxima para la cola, es decir el número de paquetes que esperan pasar de una red a otra, si el tamaño de la cola es violado el buffer de la pasarela se saturará y los paquetes adicionales se descartan en la red. Para poder recobrarse de una situación como ésta el protocolo IP necesita de un protocolo de transporte de mayor nivel, por ejemplo TCP. Es por esto que estos protocolos se usan de forma conjunta, para formar el protocolo conocido como TCP/IP.

El protocolo IP tiene como propósito hacer que el usuario no tenga que preocuparse por las características de la sub-red de comunicaciones. Este aspecto del IP es lo que lo hace atractivo, porque permite conectar redes de diferentes tipos mediante una pasarela IP.

[3] El término internet y el término Internet son diferentes, ver en el capitulo siguiente "Internet y RedUNAM" la diferencia entre estos dos términos. Por eso escribimos internet con minúscula, abandonando la convención que usábamos para remarcar las letras a partir de las cuales se forman las siglas.

Por ser un protocolo no orientado a conexión, el IP no tiene mecanismos para corrección de errores, ni para control del tráfico en la red. Los paquetes pueden perderse, o ser duplicados y por lo general no llegan en orden.

Una de las características más importantes del protocolo IP es que soporta la fragmentación de los paquetes. La sub-red de comunicación le entrega al protocolo IP un paquete de datos en base a su protocolo, IP se encarga de dividir este paquete en unidades más pequeñas. Para distinguir entre los paquetes generados por la sub-red de comunicaciones y los hechos por IP, se denomina a los paquetes de la red como Unidades de Datos (Protocol Data Units PDU) y a los paquetes creados por IP se les conoce como datagramas (ya que es un protocolo no orientado a conexión). La habilidad de romper los PDU en paquetes más pequeños (datagramas) hace al IP capaz de manejar diferentes tipos de redes, sin esta propiedad sería muy difícil manejar toda la gama de PDUs que los diferentes tipos de redes establecen. Sin esta habilidad las pasarelas tendrían que preocuparse de manejar tamaños de PDUs incompatibles entre redes. IP resuelve este problema estableciendo las reglas de fragmentación en las pasarelas y encargando el re-ensamblado de la información en las estaciones.

Por lo anterior podemos decir que la división de los PDUs en datagramas provee la transparencia necesaria entre IP y la sub-red de comunicaciones. Una vez que la red emisora a entregado los paquetes a la pasarela IP, se libera del control de la información dejando toda la responsabilidad a la pasarela. Entonces, la pasarela de acuerdo con el destino de la información determina la ruta que deben seguir los datagramas para alcanzar su destino.

Cabe aclarar que la pasarela IP no ignora por completo el protocolo que usa la sub_red de comunicación a la cual se halla conectada, la pasarela debe saber como comunicarse con la sub-red. Por lo tanto, debe existir una cierta comunicación (aunque limitada) entre la pasarela y la sub-red. Lo importante es que gracias al protocolo IP no tiene que preocuparse por las operaciones dentro de la red. Para trabajar el protocolo IP necesita del uso de direcciones. Las direcciones son números encargados de identificar de manera única a las diferentes máquinas de la red, pueden compararse con los números telefónicos. El protocolo IP establece un formato especial de 32 bits para las direcciones, su estructura se muestra en a la figura 3.12.

La dirección Internet se forma mediante la suma de la dirección de la red con la dirección de la máquina.

dirección IP = dirección de la red + dirección de la máquina

Como puede observarse la dirección no identifica a la máquina en si, sino a la máquina de acuerdo a su posición en la red; por lo tanto, si se cambia la máquina a otra red su dirección debe ser modificada.

Las direcciones IP se clasifican en clases que van desde la "A" a la "D". Como se muestra en la figura 3.12, los primeros bits de la dirección especifican la clase de la dirección, el formato de los bits restantes estará en función de la red y los subcampos que deba tener la máquina. A la dirección exclusiva de la máquina se le conoce como dirección local.

La clase A se usa en redes que tienen una gran cantidad máquinas conectadas. El campo correspondinete a la máquina tiene una longitud de 24 bits. Por lo tanto puede direccionar 2^{24} máquinas, los 7 bits restantes se usan como identificador de la red, lo que nos da la posibilidad de identificar 127 redes (la dirección cero no se utiliza como identificador para redes).

El esquema clase B se usa para redes de tamño intermedio. Se utilizan sólo catorce bits para direccionar las máquinas en la red, y los otros dieciseis bits se usan para la identificación de las redes. Las redes en esquema clase C tienen menos de 256 máquinas (2^8). Se utilizan 21 bits para la identificación de la red. Finalmente el esquema clase D se usa para transmisiónes multipunto.

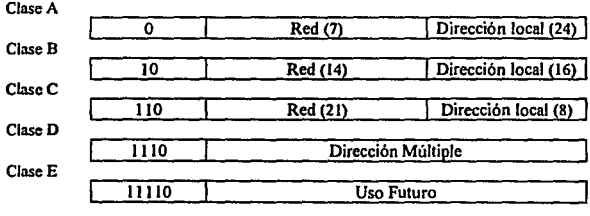


figura 3.12 Tipos de direcciones

Por convenencia las direcciones Internet son cosntruidas en fromato decimal. Por ejemplo, una dirección Clase B que en binario equivaldría a 01000000 00000011 00001001 00000001 equivaldría a 128.3.9.1. De donde la dirección de la red sería 128.3 y la de la máquina 9.1.

Las direcciones Internet en notación decimal, pueden ser las siguientes:

A red.máquina.máquina.máquina

B red.red.máquina.máquina

C red.red.red.máquina

D no aplicable

3.3.2 Protocolo de Control de Transmisión (TCP).

Algunas aplicaciones requieren asegurarse que todos los paquetes lleguen a su destino, o inclusive puede ser que la persona que efectuó la transmisión necesite asegurarse que el mensaje llego a su destino completo. Los mecanismos encargados de estas funciones se encuentran implementados en el protocolo TCP. Una de sus principales ventajas es su gran similitud con el protocolo de transporte del modelo OSI, (muchas de las características de TCP se incorporaron al estándar en el protocolo de transporte clase 4 "TP4").

El trabajo de TCP es sumamente complicado, debe satisfacer una gran cantidad de requerimientos de diferentes aplicaciones y además debe ser capaz de acomodarse a un ambiente dinámico para operar dentro de la interconexión de redes. Debe establecer y controlar sesiones entre los usuarios locales y sus contrapartes en máquinas remotas. Esto significa que TCP debe estar alerta de las actividades de los usuarios para realizar las transferencias de información que sean necesarias.

El protocolo TCP se corresponde con la capa de transporte del modelo OSI. Se encuentra por arriba de IP y por debajo de los protocolos mayores como Telnet, FTP, SNMP y NFS.

TCP es un protocolo orientado a conexión, por lo tanto establece la comunicación entre dos máquinas creando un circuito virtual. La entrega confiable de la información se establece de la siguiente manera: Primero toma la información que se desea enviar y la divide en segmentos, después enumera cada segmento para que el receptor pueda verificar que la información está completa y ponerla en el orden adecuado. Para que el protocolo TCP pueda enviar sus paquetes a través de la red, cuenta con su propio sobre[4] en el cual pone la información necesaria para el manejo de los paquetes. Cada sobre contiene un segmento de la información a transmitir. Este sobre es puesto, a su vez, dentro del sobre del protocolo IP y posteriormente es transmitido a la red. Una vez que se pone algo en un sobre IP, la red lo puede transmitir.

Del lado del destinatario el modulo receptor TCP, reúne los sobres, extrae la información de ellos y la pone en el orden adecuado. Si algún sobre se pierde en la transmisión, el receptor solicita su retransmisión al emisor. Como los mensajes pueden resultar dañados durante la transmisión de la información, (los paquetes pueden sufrir modificaciones por señales de ruido en el medio de transmisión) TCP usa rutinas de verificación para detectar errores en el mensaje. Si el mensaje está completo y no existen errores el receptor envía al transmisor una señal para informale que el mensaje fue recibido (ACK). Si el mensaje de confirmación (ACK) del receptor se perdiera o sufriera un retraso el transmisor retransmite el paquete del cual espera la confirmación de llegada.

[4] Se usa el término de sobre, comparándolo con el sobre en el cual se encuentran contenidas las cartas, no debe confundirse con el ejemplo dado para explicar los datagramas. Usamos el término de sobre porque nos pareció un buen ejemplo para determinar encapsulamieto.

Esto puede ocasionar que algunos paquetes sean duplicados; en caso que el transmisor retransmita un paquete no solicitado, el modulo TCP receptor puede descartarlo.

Otra característica importante de TCP, es que el modulo receptor puede controlar el flujo de información, para evitar que un transmisor muy rápido puede bloquearlo. Para realizar este control, TCP cuenta con una "ventana", que es el número de bytes que puede enviar el transmisor. Mientras la ventana no se encuentre llena el transmisor puede seguir enviando información, una vez que la ventana está llena el transmisor debe esperar a que halla espacio para poder continuar transmitiendo.

Para facilitar el multiplexado y permitir sesiones multiusuario dentro de una misma máquina despachadora, se usa una convención para los puertos y para otras entidades lógicas conocidas como enchufes (*sockets*). Los puertos son números asignados a cada aplicación que utiliza TCP; de acuerdo al número de puerto las máquinas pueden identificar que programa de aplicación debe recibir el tráfico que viene en camino. El uso de los puertos permite a una misma máquina ser a la vez despachadora y cliente ayudando además, a brindar servicio a muchos usuarios al mismo tiempo, etc. Las entidades lógicas conocidas como enchufes (*sockets*) también tienen asociado una dirección. La dirección del *socket* se forma mediante la concatenación del número de puerto con la dirección IP de la máquina. Este número debe ser único a través de la red para evitar ambigüedades. A continuación se muestra como se formaría la dirección de dos *sockets* uno transmisor y otro receptor:

Socket transmisor = Dirección IP origen + Número de puerto origen.
Socket receptor = Dirección IP destino + Número de puerto destino.

El modelo Internet (TCP/IP) se parece al modelo OSI, sólo que mientras el primero esta formado por cuatro niveles o capas el segundo está formado por siete. En la figura 3.13 se muestra la relación entre ambos modelos.

<table>
<tr><th colspan="2">Niveles Internet</th><th>Niveles OSI</th></tr>
<tr><td colspan="2" rowspan="3">Aplicaciones</td><td>Aplicación</td></tr>
<tr><td>Presentación</td></tr>
<tr><td>Sesión</td></tr>
<tr><td>Transmission Control Protocol (TCP)</td><td>User Datagram Protocol (UDP)</td><td>Transporte</td></tr>
<tr><td colspan="2">INTERNET PROTOCOL (IP)</td><td rowspan="2">Red</td></tr>
<tr><td colspan="2">INTERFACES</td></tr>
<tr><td colspan="2" rowspan="2">MEDIAS</td><td>Enlace de datos</td></tr>
<tr><td>Físico</td></tr>
</table>

Figura 3.13 Relación entre los modelos OSI e Internet

A pesar de ser un protocolo muy bien hecho no siempre es conveniente usar TCP, esto se debe a que establecer una conexión por medio de este protocolo requiere de una buena cantidad de trabajo y tiempo extra. Cuando no es crucial garantizar la integridad de la información es mejor usar UDP. Por ejemplo, el protocolo de procesamiento remoto (Remote Procedure Call RPC) utiliza UDP en lugar de TCP para realizar sus tareas. Este protocolo sirve simplemente como una interfase para IP. Como no tiene mecanismos para asegurar confiabilidad o control de flujo, sólo sirve como multiplexor/demultiplexor para el tráfico proveniente del protocolo IP. UDP también usa el concepto de puerto para dirigir los datagramas hacia la aplicación adecuada. El datagrama UDP contiene un puerto origen y uno destino usados para la entrega de la información.

3.4. Protocolos Superiores

Los protocolos IP y TCP se encargan garantizar la comunicación entre máquinas, forman la base para poder desarrollar programas de aplicación en red. Usando los mecanismos brindados por estos protocolos, se pueden generar aplicaciones. Dentro de este campo de aplicaciones existen algunas ya muy difundidas y perfectamente establecidas, dentro de las cuales se encuentran la transferencia de archivos, la conexión remota, el correo electrónico, etc. Para que estas aplicaciones puedan ser brindadas en cualquier máquina deben estar construidas sobre un estándar. En la presente sección analizaremos los protocolos estándar ocupados por algunas de estas aplicaciones.

3.4.1 Protocolo Telnet

El término Telnet se usa tanto para referirse al servicio como al protocolo que se usa para brindar este servicio, para diferenciar el uso de este término nosotros nos referiremos al servicio por su nombre "Telnet" y al protocolo lo llamaremos "protocolo telnet".

Telnet es una aplicación que se encarga de establecer sesiones de trabajo en cualquiera de las computadoras que forman la red Internet, hace que la computadora cliente se comporte como una terminal de la despachadora. La sesión puede ser en una máquina en la misma oficina, en la misma región o al otro lado del mundo. Este es un proyecto ambicioso, se desea conectar máquinas entre si, sin importar las características de ellas. Como los diferentes sistemas a conectar usan diferentes algoritmos para controlar pantallas, teclado y como por lo

El protocolo telnet no se encarga de la conversión entre los protocolos de la máquinas, sino que se encarga de establecer un mecanismo que determine las características de las máquinas para negociar la forma en que estas máquinas trabajarán para el intercambio de información. Este protocolo permite que la máquina cliente tenga acceso a los recursos que se encuentran el la máquina despachadora.

3.4.2 Protocolo FTP

Los estándares Internet incluyen un protocolo para la transferencia de archivos, este protocolo es llamado FTP (File Transfer Protocol) el cual es ampliamente usado en la actualidad. Con este protocolo sucede lo mismo que con el anterior como el servicio que brinda recibe el mismo nombre que el protocolo, esto puede generar confusión. Para evitar confusión haremos lo mismo que en el caso anterior, como puede notarse en el encabezado de esta subsección.

El protocolo FTP mantiene dos conexiones lógicas entre las máquinas. Una de las conexiones es utilizada para establecer el acceso, para lograr esto se vale del protocolo telnet. La otra conexión sirve para la transferencia de archivos. Este protocolo además permite abrir una sesión con una máquina hacia otra y realizar la transferencia hacia una tercera.

3.4.3 Protocolo SNMP

Existen varios protocolos diseñados para facilitar el control de la red, de estos, el protocolo de información común para la administración (Common Magnament Information Protocol CMIP) y el protocolo simple de administración de red (Simple Network Magnament Protocol SNMP) son los dos más importantes. Como SNMP se ha convertido en un estándar de facto, es el que utilizan la mayoría de las aplicaciones para la administración de red, enfatizaremos la discusión en éste.

El protocolo CIMP se basa en el modelo OSI de siete capas. Esto ocasiona que requiera mayor capacidad de procesamiento que la utilizada por SNMP (basado en TCP/IP de cuatro capas solamente). El respaldo a CMIP fue otorgado por compañías como IBM y 3Com, mediante su versión de CIMP denominada administración de redes heterogéneas (Heterogeneous LAN Magnament HLM). El objetivo de este proyecto fue concentrar la acción de CMIP sobre las dos capas inferiores del modelo OSI, para tener un sistema capaz de trabajar en máquinas con limitaciones de memoria como DOS, OS/2 y las redes de área local de computadoras personales. Este proyecto no ha tenido el éxito esperado porque el modelos OSI no ha podido afianzarse como estándar de red.

El protocolo SNMP basado en la arquitectura cliente-despachador está formado por una serie de estructuras que al interactuar fabrican las funciones necesarias para la administración de la red. Las entidades que residen en las estaciones dedicadas a la administración de la red y los elementos de la red que se comunican entre ellos usando el protocolo SNMP se conocen como "entidades de aplicación SNMP". A un par de "entidades de aplicación" con agentes SNMP se les conoce como "comunidad SNMP".

Cada comunidad es identificada mediante un nombre jerárquico dentro de Internet (no se olvide que SNMP está construido sobre TCP/IP).

Un agente es un dispositivo inteligente capaz de interactuar con el hardware de red (concentradores, ruteadores, puentes o tarjetas de red) para identificar la estructura del sistema de comunicación, colectando datos acerca de su ambiente. Los datos obtenidos por lo agentes son almacenados en una base de datos que recibe el nombre de "base de información" (Magnament Information Base MIB). El MIB contiene un conjunto estándar de variables, por ejemplo la dirección de los dispositivos y el número de paquetes IP transmitidos. Pueden establecerse dos sistemas de control, centralizado o distribuido. En el formato de control centralizado el MIB completo, con toda la información de cada uno de sus objetos, radica en sólo en una máquina. En el sistema distribuido, el MIB reside en los agentes; cada agente contiene la porción del MIB relevante a su operación.

Otra de las ventajas de SNMP sobre CIMP es que mientras el segundo sólo puede manejar objetos del tipo OSI, el primero presenta una opción para el manejo de objetos virtuales, entre ellos objetos OSI. El manejo de objetos virtuales se realiza mediante agentes apoderados. Este tipo de agente realiza las mismas funciones que los agentes normales, sólo que antes de ejecutarlas realiza una conversión entre protocolos, convierte las instrucciones SNMP de forma que puedan ser entendidas por el dispositivo propietario y viceversa.

Los mensajes SNMP son originados por las "entidades de aplicación", se considera que los mensajes pertenecen a la "comunidad SNMP" que contiene a las "entidades de aplicación". A estos mensajes se les conoce como "mensajes SNMP auténticos".

Los esquemas de identificación (authentication) se usan para verificar la autenticidad de los mensajes, con lo que se pueden implementar servicios de seguridad. A este proceso se le conoce como servicio de identificación, servicio que no se encontraba disponible en la primera versión de SNMP.

Cada elemento de la red del tipo SNMP usa un conjunto de objetos definidos en el MIB, al conjunto de objetos pertenecientes a un elemento en particular se le denomina vista del MIB. Dentro de este conjunto de objetos a cada elemento se le conoce como un modo de acceso, por ejemplo los elementos de sólo lectura o los de sólo escritura representan un modo de acceso. A la pareja formada por un modo de acceso y una vista del MIB se le conoce como perfil de la comunidad, en esencia este perfil tiene como propósito especificar los privilegios de acceso de una vista del MIB en particular. Los privilegios se establecen mediante un conjunto de archivos conocido como perfil de acceso, el cual permite establecer la forma en que los agentes y los elementos de la red pueden usar el MIB.

Para poder realizar el intercambio de mensajes, se necesita tanto de una lenguaje común como de una gramática bien definida. La información contenida en el o los MIB establece el lenguaje en común y SNMP se encarga de establecer las reglas gramaticales. El protocolo SNMP se conoce comúnmente como un protocolo de estímulo-respuesta, para cada solicitud emite una respuesta. Podemos definir tres verbos básicos en su conjunto de comandos: Get, Set y Trap. Como su nombre lo dice uno de los principales objetivos de SNMP es el formar un protocolo simple, por lo tanto usa operaciones sencillas y un número limitado de mensajes para realizar sus tareas. En el estándar se han definido cinco comandos estándar:

Get request: Este comando es usado por el administrador para accesar a los agentes y obtener información. Este comando contiene identificadores para distinguir entre peticiones sencillas y múltiples, además usa números para definir el estado de los elementos de la red.

Get next request: Este comando es similar al anterior sólo que permite pasar al siguiente identificador lógico en el árbol del MIB.

Get response: Respuesta a los comandos *get request*, *get next request* y *set request*. Contiene un identificador que lo asocia con el comando previo. Adicionalmente contiene identificadores para dar información acerca del estado de la respuesta (códigos de error, estados de error y una lista de información adicional).

Set request: Comando usado para describir una acción a ser ejecutada sobre un elemento administrado. Generalmente se utiliza para modificar valores de variables dentro del elemento.

Trap: El comando *trap* permite a un objeto de la red reportar un evento respecto al mismo o puede enviar un mensaje cambiando su estado dentro de la red.

Todas estas características han permitido a SNMP ser considerado como el protocolo por excelencia para la administración de redes. Este protocolo presenta flexibilidad, facilidad de uso e instalación, así como una cómoda interfase para poder trabajar con diversos sistemas, en diferentes ambientes.

Capítulo IV

Administración de Redes

Hoy en día los grandes avances tecnológicos nos obligan a estar mejor preparados en cada momento. El intercambio de información en forma rápida y eficiente se vuelve indispensable y el tener un sistema de comunicación confiable puede, por ejemplo, poner en ventaja a una compañía respecto a otra. Esto trae como consecuencia que se haga necesario la existencia de cierto control o administración que garantice la confiabilidad del sistema de comunicación.

Nuevamente es conveniente aclarar los conceptos que utilizaremos, ya que; por lo general existe confusión en la traducción de los términos "Network Magnament" y "Network Admnistration".

Generalmente todas las funciones que comprende el término "Network Administration" son ejecutadas por una sola persona, son tareas sencillas encaminadas a controlar una máquina o una red pequeña. Por el contrario, el término "Network Magnament" se refiere a las tareas realizadas por un equipo de trabajo para el correcto funcionamiento de un sistema completo, ya sea una red de área local o un sistema remoto, etc. Para diferenciar de alguna forma estos términos España y Argentina traducen "Network Administration" como gestión de redes, de esta forma un gestor de red es la persona que hace las veces de supervisor en las redes Novel o el super-usuario (root) de los sistemas Unix. Mientras que el término "Network Magnament" lo traducen como administración de redes. En este capitulo discutiremos la gestión de redes como una de las partes importantes de la administración de redes destacando de este último la seguridad.

Bajo el término de seguridad también se pueden diferenciar dos aspectos debido a la ambigüedad en la traducción de este término. Por un lado se entiende seguridad, como un equivalente de confiabilidad o como garantía de comunicación y por otro lado se entiende la seguridad en la red como un aspecto de privacidad de la información y de los recursos de la red. Nosotros para evitar confusiones, usaremos el término de seguridad como privacidad y el de confiabilidad como la garantía de que el determinado servicio se encuentre disponible o sea ejecutado de forma adecuada.

El problema de seguridad, puede ser observado de forma muy clara en el sistema operativo Unix, el cual es uno de los sistemas operativos más antiguo y mejor diseñado. Históricamente Unix fue creado por programadores y para ser usado por programadores. El sistema Unix era usado en un ambiente de gran cooperación, los programadores, que por lo regular tenían que trabajar de forma conjunta, preferían compartir archivos entre ellos sin tener que sortear obstáculos. Tiempo después este sistema fue adoptado por los centros de investigación de las universidades, donde existía un ambiente similar, por lo que no se necesitó implementar ningún mecanismo de seguridad.

Los problemas comienzan a principios de los ochentas, cuando Unix empieza a ser utilizado en los centros de cómputo de las universidades, en muchas compañías privadas y departamentos gubernamentales. De esta forma Unix dejó de ser utilizado en ambientes donde la cooperación entre los usuarios es el objetivo principal. Las universidades usaban a Unix y a la red para que los estudiantes realizarán sus tareas, por lo tanto no deseaban que los estudiantes pudieran intercambiar información fácilmente entre ellos.

Por otro lado las instituciones privadas que usan la red para realizar tareas confidenciales como el manejo de las cuentas de la compañía o el pago de honorarios de sus trabajadores; y las agencias de gobierno que usan la red para llevar a cabo tareas no clasificadas, con propósitos aún más sensibles, también adoptaron Unix como sistema Operativo. Estos nuevos campos de aplicación de Unix , y de las redes por lo tanto, crearon la necesidad de un sistema de seguridad.

4.1 Administración de redes

En sus inicios la administración de redes se concebía como un programa que se encargaba de una estación de alto rendimiento, una estación de trabajo Sun o HP. El programa sólo se encargaba de desplegar una serie de gráficas, útiles para monitorear el funcionamiento de la máquina, y en caso de alguna falla emitía una señal de alarma. El crecimiento de las redes generó la necesidad de un plan completo de control de la red más que un simple programa de computadora, estos programas se siguen utilizando, pero sólo como una herramienta de la administración de redes.

Sería muy difícil dar una definición precisa de lo que es la administración de redes, porque es un proceso que abarca muchas otras actividades; sin embargo, podemos decir que el administrar una red es tomar las medidas necesarias para maximizar la eficiencia y productividad de la misma. Por otro lado, Nathan J. Muller[1] señala que la administración de redes es un término que depende del contexto, con lo cual se explica la confusión que existe respecto a la traducción del concepto, que señalábamos al inicio del presente capítulo.

Para el administrador de una red de área local, el término administración se refiere a la habilidad de configurar puertos en los servidores, para controlar el acceso a las bases de datos distribuidas, puentes, pasarelas o para dar acceso a los dispositivos compartidos por los usuarios de la red (impresoras o dispositivos de almacenamiento, por ejemplo). En este ambiente el funcionamiento de los servidores es crítico para el trabajo de la red, por lo tanto deben ser administrados de forma correcta. A la administración de cada servidor en particular es a lo que los españoles y argentinos llaman gestión de red.

---

[1] Nathan J. Muller es un consultor independiente en Oxford, se especializa en tecnología enfocada a la comercilización y la educación. Es coautor del libro "LAN's to WAN's: Network Magnament en 1990's".

Por otro lado, para el administrador de una red de área amplia, el término de administración significa hecerse cargo de revisar el funcionamiento de las líneas de comunicación para asegurar la integridad de los datos. En este aspecto la administración de la red incluye la habilidad de tomar las medidas apropiadas para restaurar equipo que se encuentre fallando o en su defecto, buscar métodos para evitar las zonas de falla antes que los problemas degraden el funcionamiento de la red. Y de esta forma podemos localizar diferentes ambientes, donde la administración de red implicará la ejecución de tareas diferentes.

La importancia de la administración de una red radica en las enormes ventajas que una empresa puede obtener de esta disciplina. El propósito del establecimiento de una red no es sólo tener una forma de comunicación más, sino obtener de ellas la productividad y eficiencia que hagan de sus empresa competitivas. Por lo tanto el gasto realizado, ya sea en mantener un equipo encargado del correcto funcionamiento de la red o de el pago a una compañía especializada, permite a la empresa contratante aumentar su rendimiento y funcionalidad.

Para ayudar a la administración de la red los dispositivos para la interconexión de redes, puentes, ruteadores y pasarelas, están equipados con capacidades de control y reporte de funcionamiento. Los concentradores ya no son sólo puntos de unión entre los diferentes cables, empiezan a ser más inteligentes, es decir soportan una gran cantidad de funciones de administración de red.. Con estas herramientas para la administración de la red, los técnicos pueden diagnosticar y corregir problemas inclusive de forma remota. Existen además, muchos dispositivos que se encargan del diagnostico y control de las redes, pero aun no es posible que todos estos dispositivos sean integrados en un punto central de control. A pesar de la construcción de los estándares, los sistemas abiertos no pueden ser manejados como si fueran un

sistema homogéneo. Los enlaces de las redes de área local con las de área metropolitana han creado un nivel de complejidad que estos dispositivos todavía no son capaces de manejar. Inclusive en redes propietarias, se requiere de diferentes sistemas de control, esto complica la tarea de administrar una red.

La tarea de tener una red operando de forma armónica es un reto enorme. Algunas compañías dependen de los fabricantes de los sistemas para corregir los problemas de sus redes; otras contratan los servicios de compañías especializadas en la solución de problemas de comunicación. La forma de administración por medio de una compañía que se encargue de la administración de la red, ya sea parcial o total, se está popularizando en los Estados Unidos, sobre todo en las compañías que tienen problemas económicos. Contratar, entrenar y mantener un grupo de técnicos especializados, se está convirtiendo en un lujo que sólo unas cuantas compañías pueden soportar. En México la situación no es diferente, existen ya grandes compañías y corporaciones que tiene redes instaladas: PEMEX, Comisión Federal de Electricidad, Estafeta, Comisión Nacional Bancaria, etc; pero no tienen el personal capacitado para su administración, lo que genera un campo de trabajo que podría ser explotado por nosotros, los ingenieros del país.

4.2 Marco de trabajo para la administración de redes OSI

La administración de una red es una tarea que abarca muchas áreas, por lo tanto podemos subdividir la administración de la red de acuerdo a sus campos de funcionamiento. De hecho, esto fue realizado por la Organización Internacional de Estándares (ISO) con la definición de un modelo en base a cinco áreas funcionales para la administración de una red.

4.2.1 Administración de configuración de la red

La administración de la configuración de la red incluye tanto el control de la configuración actual como el de posibles cambios en la configuración. Las tareas de este nivel incluyen la recopilación detallada de toda la información referente a los elementos que conforman la red, tanto de software como de hardware. Se debe conocer su ubicación, características, números de serie y versiones.

Las redes complejas, con cientos o miles de dispositivos, deben tener un sistema de administración que opere bajo el control de software especializado que facilite el control de la red desde un sólo punto. Este sistema probablemente se encargará de mostrar una representación gráfica de la red, además de otorgar al usuario la habilidad de leer y cambiar los parámetros de los dispositivos mediante parámetros de línea. En estos sistemas por lo general se cuenta con varias líneas de transmisión y el ajuste de los parámetros del sistema, normalmente, se deja a decisión del administrador. De esta forma, la habilidad de obtener rápidamente los parámetros de línea permite conocer el estado de todas las vías de comunicación, para que el administrador pueda determinar caminos alternativos para dirigir la información a través de la red, en caso de la falla de alguna de las líneas de transmisión.

4.2.2 Administración de fallas

La administración de fallas se refiere al proceso de detección, aislamiento, manejo y la eventual solución de las fallas. Este servicio es crítico por el costo que implica tener al sistema fuera de operación. El paso principal de la administración de fallas es detectar los errores en la red. Esto puede ser realizado por una variedad de métodos. Por ejemplo, se puede establecer una

alarma que se activa al sentir determinado comportamiento en el funcionamiento de la red, acto seguido es necesario revisar el lugar donde se generó la alarma para ver si existe o no un problema. La solución de los problemas dependerá de la política establecida por la compañía dueña de la red, se puede tener un equipo de trabajo encargado a la compostura de desperfectos o bien se puede reportar los equipos que presentan la falla a su fabricante para que este repare el problema.

4.2.3 Administración de rendimiento

El rendimiento del sistema envuelve a las tareas que deben ser ejecutadas para evaluar el grado de utilización del equipo de la red, con el objetivo de detectar sobrecargas o cargas muy bajas de trabajo que afecten el buen funcionamiento del sistema. Se analizan también las zonas donde el tráfico tiende a crecer. De esta forma se puede estar un paso adelante de las necesidades actuales y futuras de la red. Con la administración del rendimiento se logra eliminar una de las características más comunes en las redes instaladas y es que su crecimiento no fue necesariamente ordenado, lo que ocasionó un desbalanceo en las cargas de trabajo entre servidores, grupos de trabajo o sectores de la red. Con un buen análisis del rendimiento de la red puede incrementarse el rendimiento del sistema.

Los problemas de exceso de capacidad tienen que ser detectados por el personal a cargo de la administración de la red, por lo que este personal tiene que examinar el potencial de la red tanto para su expansión como para su posible contracción.

4.2.4 Administración de seguridad

La administración de seguridad se encarga de asegurar que sólo personal autorizado haga uso de la red. Las tareas o funciones asociadas con la administración de seguridad son la identificación de posibles puntos de acceso no autorizado, cifrado de los datos, la administración de las claves para cifrar datos, administración de las claves de acceso y la verificación de la seguridad de las mismas. Como parte de la administración de seguridad se incluyen las tareas encaminadas a la prevención de ataques de virus, procesos que garanticen la continuidad de operación y la planeación de sistemas encaminados a recuperar la información en caso de que se presente algún desastre. Aunque los administradores de la red no pueden estar como guardias de la red, para evitar la transmisión de archivos sospechosos, pueden y deben publicar métodos para probar el software desconocido así como los procedimientos a seguir para la obtención de software público.

4.2.5 Administración de costo de uso de la red

Uno de los procesos de administración de la red consiste en obtener información que permita establecer la cantidad de dinero que se debe cobrar a los usuarios por concepto de utilización de los recursos de la misma. Por medio de esta parte de la administración se pueden determinar las áreas de mayor costo de la red y si este costo tiene sentido con las áreas que lo generan. Las tareas de administración de costo incluyen la expedición de ordenes de pago y la recolección de facturas, el cálculo de los cargos de amortización y depreciación y la asignación de costos por persona de uso de la red mediante el desarrollo de algoritmos que cobren a los

usuarios proporcionalmente a su uso de los recursos de la red y por último la continua revisión de los métodos de cobro para asegurarse que son justos y equitativos.

El proceso de administración de costos puede requerir de los esfuerzos de un grupo de especialistas, en organizaciones de gran tamaño. Para organizaciones de medianas y pequeñas, el esfuerzo que comprende la administración de costos puede ser todavía considerable, especialmente en comparación con las otras funciones de la administración de red. Muchas organizaciones tienen un cuota fija para cada uno de los usuarios, este procedimiento reduce las actividades asociadas con la administración del costo, de esta manera no es necesario registrar la continuidad de uso de los sistemas por cada uno de los usuarios. De cualquier forma la administración de costo de la red sigue siendo una de las partes principales de la administración de la red.

4.2.6 Software para la administración de red

La importancia de la administración de redes y su complejidad han ocasionado que los fabricantes de software estén interesados en el diseño de sistemas capaces de facilitar las tareas de administración de una red. Por lo general estos sistemas están construidos para operar por medio del protocolo SNMP, debido a la flexibilidad y transportabilidad de este protocolo. Como ejemplo de un sistema de este tipo analizaremos las caracteristicas más importantes del sistema manejador de red "Open View" fabricado por la compañía Hewlett Packard.

Este sistema está diseñado para manejar redes TCP/IP ya que está basado en el protocolo SNMP. La instalación de Open View se realiza en tres pasos, primero se necesita instalar OVIC el programa que se encargará de la instalación de Open View y el que además contiene la clave de

activación, requerida para el uso de Open View. Después de instalar OVIC se debe instalar la plataforma SNMP de Open View para continuar con la instalación de Open View propiamente dicha.

Este sistema no tiene problemas para reconocer, de forma automática, la mayoría de los dispositivos de red que usan agentes SNMP, pero dispositivos con versiones antiguas de este protocolo pueden presentar problema para ser detectados de forma automática.

Las computadoras personales y Macintosh puede que no sean detectadas de forma automática, para que Open View pueda detectarlas automáticamente deben estar encendidas al momento de realizar la búsqueda. Adicionalmente a la localización automática de nodos Open View puede descubrir la topología de una red mediante un archivo que contenga los nombres de las diferentes máquinas, el archivo */etc/hosts*, por ejemplo. El resultado obtenido durante el rastreo de la configuración de la red se presenta al usuario mediante una interfase gráfica, cada nodo se representa por medio de un icono.

Open View presenta un interfase bastante amigable en base a iconos y ventanas, construida sobre Motif, tiene una gran cantidad de menús y submenus para interactuar con el sistema. Se pueden crear nuevas ventanas y menús de acuerdo a las necesidades particulares, etc. La mayor parte de la configuración de la red es presentada por medio de gráficas (mapas) e iconos, pero puede presentar información adicional por medio de los menús. Se puede obtener información adicional de cada nodo, fabricante del nodo, características de la tarjeta Ethernet, la localización física del nodo, a quien se deben reportar los problemas correspondientes a un nodo determinado, etc.

Este sistema para la administración de red es un buen esfuerzo; sin embargo, presenta ciertas limitaciones, las principales son: para ejecutarlo se requiere de cierto equipo de hardware así como cierta configuración del sistema operativo; no puede ser usado en sistemas multiprocesardor; está garantizado para trabajar correctamente en la versión actual de Solaris (SunOS 4.1.x), pero no podrá ejecutarse en sistemas con Solaris 2.x; los requerimientos de memoria, tanto RAM como de Swap, están basados en la red y el número de sesiones SNMP que sea necesario mantener. Estos requerimientos hacen que se necesite un sistema dedicado exclusivamente a esta aplicación.

4.3 Administración de una máquina Unix[2]

Como usuarios de sistemas personales estamos acostumbrados a trabajar en un sistema mono-usuario, el de las computadoras personales, que a pesar de tener sus limitaciones resultan ser muy amigables y fáciles de utilizar. La mayoría de las computadoras personales usan el sistema operativo MS-DOS, que se caracteriza por estar construido para ser accesible al usuario inexperto. Por el contrario los sistemas Unix tienen fama de ser más complicados. Esto resulta cierto sólo parcialmente. En realidad Unix es más sencillo que otros sistemas operativos, debido a su filosofía de lo pequeño es bello, paradójicamente esto es lo que hace que se le considere complicado.

Las funciones del administrador de una máquina Unix son numerosas y variadas; sin embargo, podemos identificar algunas tareas típicas:

---

[2] Por tratarse en particular del control de una máquina Unix nos tomamos la libertad de tomar el término de administración, porque no es más familiar. Sin embargo, en términos estrictos para seguir con nuestra notación deberíamos usar el término de gestión. Esperamos que esta corrección en la notación facilite la lectura de esta sección en lugar de dificultarla.

El ser un sistema multi-usuario hace que Unix necesite de alguien que coordine el trabajo de los diferentes usuarios, el administrador debe asegurarse que las operaciones que ejecuten los usuarios no perjudiquen el trabajo de los demás. Debe de encargarse de registrar a los nuevos usuarios del sistema y dar de baja a los que ya no deban tener acceso a éste. Para poder registrar a un nuevo usuario se le debe asignar un nombre y decidir la colocación de su directorio "HOME". Cuando un usuario deja de ser miembro activo, el administrador debe eliminar su registro y sus archivos.

Los sistemas Unix tienen que ser configurados para poder usar las diferentes piezas de hardware de las que estén formados. Existen sistemas que tienen piezas de hardware sofisticadas que nunca pueden ser utilizadas ya que nadie se tomó la molestia de configurarlas. De igual forma cuando alguna pieza de hardware se remueve del sistema, este debe ser reconfigurado para evitar problemas. Por lo tanto el administrador debe encargarse de la configuración del sistema.

Otra de las tareas del administrador es respaldar la información de los usuarios, está es una de las tareas más importantes de la gestión de una máquina Unix, si algo es valioso para los usuarios es su información. Por lo regular esta resulta ser una tarea lenta y aburrida, por lo que muchos tratan de restarle importancia. Aunque puede ser automatizada, es responsabilidad del administrador verificar que ha sido ejecutada de forma correcta.

Además, en un sistema mono-usuario el dueño de la computadora decide si instala o no determinado programa en su computadora, pero en un sistema multi-usuario alguien se tiene que encargar de la instalación del software de utilidad en la máquina o de lo contrario los dispositivos de almacenamiento de la máquina se saturarían, ya que podría darse el caso de tener software duplicado. Para evitar estos problemas el administrador debe encontrar el software que sea de

mayor utilidad a los usuarios de su sistema, instalarlo e informar a los usuarios de su disponibilidad, colocación y forma de acceso. El software local debe colocarse en un lugar donde sea fácilmente diferenciable del software incluido con la versión de Unix en uso, de esta manera será más fácil la instalación de nuevas versiones de Unix

Para cerciorase del correcto funcionamiento de la red, el administrador debe revisar continuamente el funcionamiento de la misma. Todos los días debe revisar que el sistema de correo electrónico esté trabajando adecuadamente, que la conexión hacia otras máquinas esta disponible, etc. Además debe estar en contacto con los usuarios para que estos puedan reportar las fallas y para servirles como apoyo, asesorarles, etc.

Para hacer más sencilla la tarea de administrar un sistema, el administrador debe crear la documentación necesaria para conocer la composición de su sistema. Es responsabilidad del administrador documentar todos los aspectos del sistema que se refuerzan a su ambiente local. Esto incluye documentar cualquier software instalado, el tipo de cables instalados y su función, registros del mantenimiento del hardware y registros del estado de los respaldos.

El administrador se debe encargar de establecer una política de seguridad y periódicamente revisar si el sistema de seguridad no ha sido violado.

Es también conveniente establecer un sistema de contabilidad, para revisar el uso que dan los diferentes usuarios a los recursos del sistema. Esto resulta indispensable en sistemas donde se le cobra al usuario el uso de los dispositivos del sistema, pero también resulta útil conocer la forma de utilización de los dispositivos independientemente del cobró o no por su uso.

4.3.1 Administración del acceso a una máquina Unix

El establecimiento de los permisos de un archivo y el concepto de pertenencia son de gran utilidad para evitar que los archivos de los usuarios particulares sean modificados sin autorización por otros usuarios, pero el administrador tiene que romper estas barreras de protección sobre los archivos para garantizar el control sobre el sistema. Unix establece esto tratando al administrador como un usuario especial, debido a los privilegios que otorga Unix a este usuario se le conoce como el super-usuario (superuser) o usuario raíz (root)

El super-usuario tiene derecho a ejecutar cualquier operación sobre los archivos, de la misma forma existen llamadas al sistema que sólo pueden ser ejecutadas por el super-usuario. Debe tenerse cuidado con las operaciones que se ejecutan como super-usuario, ya que el resultado de estas pueden afectar el funcionamiento del sistema.

Un usuario no es necesariamente una persona, puede ser un programa que se ejecuta de forma automática. Es necesario tener esto en mente cuando se establecen medidas de seguridad. Para dar de alta a un usuario, sin importar si es un individuo o un proceso, el administrador debe asignarle un nombre[3], el nombre por el cual el usuario será conocido dentro del sistema. Para Unix no es importante el nombre en si mismo, lo que Unix maneja es un número asociado al nombre. A este número se le conoce como número de identificación del usuario (UID number), el sistema se encarga de establecer este número automáticamente.

Para distinguir entre los usuarios reales y los procesos, estos últimos tienen asociados UIDs mayores que uno y menores que 100 mientras que los primeros tienen asociados números

[3] A los nombres de los usuarios en el sistema se les conoce con el nombre genérico de *login*, ya que es la identificación que usan para presentarse al sistema. No intentamos la traducción de este término ya que está ampliamente aceptado entre los usuarios, de esta manera es común oír que un usuario le pregunta a otro ¿Cuál es tu login?

entre 100 y 32767. El UID cero esta reservado para el administrador o super-usuario. Además de asignarle un *login* el super-usuario necesita determinar a que grupo pertenecerá el usuario. El número de grupos dentro de un sistema dependerá de las necesidades del sistema en particular, en el archivo */etc/group* se encuentra una lista de todos los grupos del sistema.

Para evitar el acceso no autorizado al sistema cada usuario tiene una contraseña (password) que sirve para probar su identidad al sistema, ya que los nombres de los usuarios son públicos. Cualquiera puede conocer los nombres de los usuarios del sistema, pero nadie puede conocer la contraseña de los usuarios, ni siquiera el super-usuario, a menos que el usuario la publique. Para manejar las contraseñas de los usuarios Unix tiene el archivo */etc/password*, el cual contiene el nombre del usuario y una versión cifrada de su contraseña. De esta forma cada vez que se anexa un usuario al sistema el administrador tiene que asignarle un espacio dentro del archivo */etc/password* para que Unix pueda darle acceso. Por medidas de seguridad el administrador otorga una contraseña al usuario al momento de registrarlo, después le informa al usuario su login y contraseña para que pueda accesar el sistema. Una vez en el sistema el usuario puede cambiar su contraseña, no existe ninguna razón por lo cual el administrador debe conocerla.

Como una medida para organizar la información, cada usuario tendrá un subdirectorio asociado, a este subdirectorio se le conoce como directorio *home*, por lo general el subdirectorio *home* tiene el mismo nombre que el *login* del usuario para que a simple vista pueda determinarse que tales archivos pertenecen a determinado usuario. El administrador debe otorgar los derechos correspondientes al usuario sobre su directorio *home* y sobre sus archivos de inicialización. Los

archivos de inicialización son archivos que permiten personalizar el comportamiento del sistema, además, dentro de estos archivos su copia del shell.

Para terminar el administrador debe crear un archivo especial, *mail* en el directorio Mail, para que el usuario pueda hacer uso del correo electrónico. Es conveniente que el administrador pruebe la cuenta antes de otorgarla al usuario de esta manera podrá solucionar cualquier error que se haya ejecutado al momento del registro.

Como esta es una tarea rutinaria y como siempre que se da de alta a un usuario deben ejecutarse los mismos pasos existen numerosos programas que facilitan esta labor, e inclusive el propio super-usuario puede definir uno de acuerdo a las características de su sistema en particular. Adicionalmente la mayoría de los sistemas actuales tienen un programa interactivo que facilitar en gran medida este tipo de tareas. Un ejemplos de este tipo de programas lo tenemos en el programa *sam* que contiene la versión Unix de HP, *sam* permite realizar la mayoría de las tareas rutinarias de la administración de una forma muy sencilla, sin embargo es aconsejable que el administrador conozca las cosas que deben realizarse para la ejecución de las tareas para que pueda intervenir en caso de que se presenten problemas.

El dar de baja a un usuario del sistema es en teoría un proceso más sencillo que el darlo de alta. Todo lo que se tiene que hacer es cancelar su identificación dentro del archivo */etc/password* y remover toda su información del sistema de archivos. Es conveniente respaldar la información del usuario antes de darlo de baja y también es de suma importancia que este usuario no tenga acceso al sistema a menos que el administrador se lo vuelva a otorgar.

4.3.2 El sistema de archivos visto por el administrador

Unix es un sistema que se caracteriza por tener pocos conceptos para trabajar con diferentes componentes. Así todo lo relativo a entrada y salida se maneja por medio de archivos, la información referente al sistema se encuentra en numerosos archivos de configuración, algunos de los comandos se basan en la información contenida en ciertos archivos para su funcionamiento, etc. Por lo tanto una de las partes más importantes de Unix es su sistema de archivos, desgraciadamente este es también una de sus partes más vulnerables. Existen una gran cantidad de formas en que se puede dañar el sistema de archivos de Unix; sin embargo, la causa más común es la terminación abrupta de la ejecución del mismo; esto puede suceder por una pérdida de energía, por apagar la computadora sin ejecutar el debido procedimiento o por un error interno en el núcleo (kernel).

Para revisar el sistema de archivos el administrador cuenta con el comando *fsck*. Este comando se encarga de revisar el sistema de archivos en base a la información contenida en el archivo, /etc/fstab en los sistemas BSD o /etc/checklist en sistemas AT&T.

El arreglar el sistema de archivos a mano, es una operación que requiere de gran experiencia del usuario; por lo tanto, no es recomendable hacerlo. Por lo tanto cuando el sistema de archivos sufra algún desperfecto lo más recomendable es usar *fsck* para arreglarlo, ya que este comando no sólo revisa el sistema de archivos sino que también es capaz de realizar las reparaciones. El comando *fsck* trabaja de forma interactiva, por lo tanto antes de realizar una reparación enviará un mensaje preguntado si se desea corregir el error o no. Por lo general, no existe razón para negarse a aceptar la ayuda de *fsck*.

4.3.3 Apagado de una máquina Unix

Para apagar una máquina Unix es necesario realizar una serie de procedimientos que son ejecutados por el administrador de la máquina.

Generalmente los pasos a seguir para poder apagar el sistema son los siguientes: advertir a los usuarios (para que puedan realizar las operaciones pertinentes, salvar su información, etc), suspender los procesos en ejecución (esto se realiza mediante el mandato kill), desmontar los sistemas de archivos que hayan sido montados en el sistema de archivos raíz, poner al sistema en modo mono-usuario y usar los comandos *sync* para copiar los archivos residentes en la memoria al disco. La mayoría de los sistemas cuenta con un programa para descargar el sistema creado por el fabricante. Este programa por lo general tiene la siguiente ubicación: */etc/shutdown* o */sbin/shutdown*. Este programa tiene un modificador para otorgar un periodo de gracia a los usuarios antes de apagar el sistema. Conforme el periodo de gracia se va agotando, envía mensajes a los usuarios para concientizarlos. Para apagar el sistema sin periodo de gracia el comando *shutdown* cuenta con el modificador *-h* (equivalente del comando *halt* de los sistemas BSD) el cual apaga el sistema sin otorgar el periodo de gracia. Como esta opción no da tiempo a los usuarios de hacer nada sólo debe usarse en caso de emergencia. Aunque, por lo general, el sistema se apaga sólo cuando los usuarios ya no tienen acceso al sistema. Una vez que *shutdown* ha terminado de ejecutar todas sus funciones envía a la pantalla un mensaje donde informa que el sistema esta listo para ser apagado físicamente.

Un proceso equivalente al de pagado es el de reiniciar la máquina. Para reiniciar un sistema Unix se necesitan efectuar los mismos pasos que en el caso anterior, sólo que en lugar de enviar el mensaje indicando que la máquina está lista para ser apagada la máquina ejecuta su

procedimiento de encendido de nueva cuenta. Como ya mencionamos es necesario reiniciar la máquina cuando queremos que algún cambio en los archivos de configuración que se ejecutan al encendido de la máquina ha sido modificado y queremos que esta modificación tenga efecto, cuando se instalan nuevos periféricos, etc. Para reiniciar la máquina los sistemas cuentan con el comando *reboot* que es equivalente al comando *shutdown -r*.

Existen fallas capaces de hacer que el sistema no pueda ser iniciado, de allí que sea recomendable tener siempre una copia del sistema para iniciar la máquina en caso de problemas. A continuación discutiremos algunos de los problemas que pueden hacer que una máquina Unix no arranque. La mayoría de los autores coincide en señalar que los problemas que pueden hacer que una máquina Unix no arranque pueden ser clasificados de la siguiente manera:

Errores de hardware

Defectos en el dispositivo de arranque (discos o cintas)

Errores de inicialización en los archivos de configuración al arranque

Errores en el kernel

Errores en el sistema de archivos

Lo primero es identificar el tipo de problema, para estos es recomendable leer todos los mensajes que el sistema envía al encenderse. Además muchos sistemas cuentan con un archivo donde guardan todos los mensajes de error, el archivo que contiene estos mensajes por lo general es */usr/adm/message*. Otros sistemas son capaces de generar una copia de la memoria al momento en el que el kernel sufrió la caída, a esta imagen del kernel se le conoce como *crash dump*. Es

necesario conocer la forma de como salvar esta imagen antes de reiniciar la máquina ya que la información allí contenida puede ser muy útil para resolver problemas.

Errores de hardware.- Como administrador no hay mucho que hacer para la solución de este tipo de errores, lo mejor es llamar a los técnicos especializados. En algunas ocasiones el sistema indica que se trata de una falla de hardware (un error de paridad en la memoria, por ejemplo), sin embargo es conveniente asegurarse que se trata realmente de una falla de hardware antes de llamar a los técnicos, hay que revisar todas las conexiones, verificar que todos los dispositivos están encendidos, etc. Es conveniente ejecutar todos los programas de diagnostico disponibles en el sistema antes de llamar a los técnicos.

Defectos en el dispositivo de arranque.- Este tipo de error es fácil de detectar, siempre y cuando se cuente con un medio alternativo para el arranque del sistema. El tener un medio alternativo para el arranque del sistema es no solamente recomendable sino necesario, conforme cambie la configuración del sistema de archivos la copia de respaldo del sistema de arranque debe ser actualizada o de lo contrario resultará inútil.

Errores en la configuración de los archivos de arranque .- Los archivos de configuración pueden resultar muy complicados por lo que es conveniente tener una copia de ellos en caso de hacer cualquier modificación. Por lo general los errores en este tipo de archivos hacen que el sistema no pueda entrar en modo multi-usuario, por lo que el error deberá corregirse en el nivel de mono-usuario, lo que tiene como desventaja de que no estarán disponibles algunas aplicaciones como vi por ejemplo.

Errores en el kernel .- En Unix inclusive el kernel puede sufrir ciertas modificaciones para adaptarse a las necesidades de los usuarios, sin embargo no es una tarea fácil realizar

modificaciones sobre el kernel. Por lo general es conveniente tener una copia del kernel antes de realizar cualquier modificación.

Errores en el sistema de archivos .- Este tipo de errores son los más peligrosos, de allí todos los cuidados que tienen que tenerse para descargar el sistema, el sistema de archivos puede dañarse de tal forma que pueda ser irrecobrable. El daño en el sistema de archivos puede ser ligero (información revuelta en el disco) o bien puede ser más serio como una falla en las cabezas lectoras del disco. Si el sistema raíz no puede ser leído no podrá cargarse el kernel y el sistema se comportará como si la falla fuera de hardware. Cuando sucede esto se puede intentar cargar el sistema a partir de otra fuente, sin embargo esto requerirá de tiempo. Si el sistema entra en modo mono-usuario se puede ejecutar el comando *fsck* sobre el sistema raíz para tratar de repararlo y montar después el resto de los sistemas de archivos, si el sistema no entra en modo mono-usuario el problema es más serio. En estos casos lo más probable será que la única solución factible sea cargar de nueva cuenta en sistema en la máquina a partir de su versión original, si se tienen versiones actualizadas de los respaldos los daños generados por la ejecución de esta tarea serán mínimos.

4.3.4 Montar y desmontar sistemas de archivos

Como administrador se debe estar consciente de que el sistema de archivos de Unix no es una entidad homogénea. Siguiendo su filosofía de modularidad, el sistema de archivos de Unix está compuesto por una serie de partes funcionales que interoperan para dar la apariencia de un sistema homogéneo. De esta forma el sistema de archivos Unix en realidad es un conjunto de sub_sistemas de archivos trabajando de forma armónica, el sistema Unix más pequeño tiene al

menos dos sistemas de archivos: el sistema Unix estándar (el cual contiene el kernel, los directorios */etc*, */bin* y los archivos de los usuarios) y un sistema de archivos especial para realizar el *swap* (este proceso se explica más adelante). Cada uno de estos sistemas se guarda en una sección diferente del disco para formar el sistema de archivos general, se dice que cada uno de los sub-sistemas se monta, el montar un sistema de archivos en otro significa establecer un enlace entre uno y otro[4]. Los diferentes sistemas de archivos se montan para crear un árbol o sistema de archivos mayor. La característica principal de cada uno de estos sistemas de archivos es que cada uno de ellos tiene su propio directorio raíz, lo que ayuda al kernel a resolver las rutas para accesar a los archivos. El usuario puede accesar a cualquier archivo de la forma convencional, sin importar si dicho archivo pertenece a un sistema de archivos montado o no. Por el contrario la información que se encuentre en sistemas de archivos no montados no puede ser accesada.

Unix es un sistema multi-usuario y por lo general es bastante grande también. Por ejemplo, un sistema Unix de una mainframe tiene que lidiar con billones de bits de información, de cientos de usuarios; que a su vez necesitan la garantía de la integridad de su información. El mantener la integridad de la información no resulta una tarea muy sencilla en sistemas multi-usuario. Esta resulta una característica importante de la tarea de montar sistemas de archivos. El hecho de montar un sistema de archivos sobre otro puede no sólo mantener separados diferentes sistemas de archivos, sino que también los mantienen inviolables, ya que Unix es bastante estricto en lo que a los límites de los sistemas de archivos se refiere. Como cada sistema de archivos se encuentra en una diferente partición del disco duro, la comunicación entre los diferentes sistemas de archivos se

[4] Como el área ocupada para el proceso de *swap* no se monta, estrictamente hablando no podemos considerar a esta parte como un sistema de archivos; sin embargo, funciona bien como ejemplo de un sistema de archivos siempre y cuando se tenga en mente este inconveniente.

encuentra rígidamente controlada por Unix. Si algún usuario trata de establecer un enlace (mediante el comando *ln*, por ejemplo) entre dos archivos en diferentes sistemas de archivos, el sistema se rehusara a realizarlo; Unix sólo permite realizar enlaces simbólicos entre sistemas de archivos. Aislando de esta manera los datos, se pude prevenir la corrupción de la información y se ayuda a establecer politicas de seguridad.

Los sistemas Unix grandes por lo general tienen grandes capacidades de almacenamiento, por lo que tienen muchos sistemas de archivos montados. En este tipo de sistemas es común encontrar demasiados sistemas de archivos para diferentes grupos de usuarios.

Los sistemas más pequeños aunque no necesitan montar muchos sistemas de archivos pueden ser más versátiles y fáciles de expandir si se dividen en unos cuantos sistemas de archivos.

El administrador de archivos debe planear su sistema de archivos, para que el sistema de archivos completo esté de acuerdo a sus necesidades. Para crear el sistema de archivos completo, se deben ejecutar básicamente dos pasos:

- Los sub-sistemas de archivos son creados individualmente, esto se hace mediante el comando *mksf*.

- Los diferentes sistemas de archivos deben montarse para formar el sistema general.

Como cada sistema de archivos se guarda en una sección diferente del disco, éste tiene que ser dividido en diferentes secciones, se debe ser cuidadoso para determinar el tamaño de cada una de las particiones; porque del tamaño de estas dependerá la eficiencia del sistema. La creación de las particiones no es manejada por Unix, por lo que algunos sistemas ya tienen realizadas algunas particiones de fabrica.

Por medidas de seguridad es recomendable desmontar los sistemas de archivos antes de ejecutar algunas tareas críticas, como el formateo o la creación de un sistema de archivos nuevo. El desmontar un sistema es tan sencillo como la ejecución del comando *unmount*. Como la ejecución de este comando hace que el sistema de archivos desmontado sea imposible de ser accesado, los sistemas de archivos no pueden ser desmontados si algún usuario está usando uno de los archivos del sistema que quiere desmontarse o si algún proceso del sistema de archivos está en ejecución. Debido a la dificultad que implica revisar que ninguno de los archivos del sistema de archivos a desmontar esté en uso, es más conveniente desmontar un sistema de archivos una vez que todos los usuarios se encuentran fuera del sistema.

Para facilitar el trabajo del administrador en lo referente a los sistemas de archivos, Unix cuanta con una serie de herramientas. Entre estas herramientas tenemos el archivo */etc/mnttab*, el cual tiene la información concerniente a los sistemas de archivos montados. Cuando el sistema sufrió una caída, el archivo */etc/mnttab* existe cuando el sistema se incorpora de nuevo, lo que ocasionará problemas ya que este archivo contendrá tanto la información anterior a la caída como la información generada durante el nuevo encendido. Para evitar problemas de este tipo es necesario revisar que el archivo */etc/rc* tenga unas líneas dedicadas a borrar el contenido de */etc/mnttab* antes de montar de nuevo los sistemas de archivos.

Además de */etc/mnttab* en la mayoría de los sistemas se encuentran archivos para la ejecución del montado y desmontado de los sistemas de archivos de forma automática, los archivos para montar y desmontar archivos de forma automática son */etc/Mount* y */etc/Unmount* respectivamente. Adicionalmente el administrador debe usar los comandos *du* y *fsstat*. El primero

sirve para determinar el espacio utilizado en el disco y así evitar problemas de saturación y el segundo sirve para revisar la integridad de un sistema de archivos antes de que sea montado.

4.3.5 Respaldos

Realizar respaldos de la información es una de las tareas más tediosas y también una de las más importantes de la administración de un sistema. Inclusive los usuarios de los sistemas personales se dan cuenta de la importancia de esta labor, la aparición de los virus informáticos concientizó a más de un usuario de una forma bastante desagradable. Aunque en Unix no existe el peligro de los virus, al menos no hasta ahora, tarde o temprano el administrador tendrá que enfrentarse con la pérdida de archivos, por lo que es mejor estar preparado.

Además de saber que se necesita respaldar la información es necesario establecer una política para determinar el período de tiempo que debe existir entre un respaldo y otro, entre más frecuentemente se realice el respaldo menos riesgo se tendrá de perder información; pero, se debe estar consciente que este proceso requiere tiempo energía y uso de recursos como discos y cintas, además que el tiempo que dure el proceso de respaldar la información el sistema estará fuera de servicio. Debido a todo esto no existe una fórmula para establecer la política de respaldo, esta política estará en función a los requerimientos del sistema.

Existen dos formas de realizar un respaldo, se puede realizar un respaldo parcial o bien un respaldo total. Como el respaldo total requerirá de más tiempo y recursos es aconsejable realizarlo sólo la primera vez que se instalo el sistema y cuando se realicen cambios significativos en el mismo, como la instalación de un nuevo disco, etc. Un respaldo parcial es una copia de un sistema de archivos o de una rama del directorio del árbol. El administrador del sistema tiene que

asegurarse de realizar respaldos parciales de los sistemas de archivos que sufren más cambios. Se pueden realizar respaldos increméntales para que sólo incluyan copias de los nuevos archivos y de los que sufrieron modificaciones.

Los respaldos se pueden realizar en diferentes medios, que van desde discos flexibles (disquetes), cintas magnéticas convencionales, cartuchos magnéticos, cintas de videos y discos gusano (worm disks). La cintas magnéticas solían ser el principal medio para realizar respaldos, sin embargo en la actualidad están prácticamente fuera de uso. Los discos flexibles son ampliamente conocidos, deben su popularidad a los sistemas personales. Los cartuchos de cinta son en la actualidad el medio estándar para la realización de respaldos, no son comunes en las computadoras personales pero si en las estaciones de trabajo donde se necesita respaldar una cantidad de información que no podría ser manejada en disquetes, las cintas tienen una capacidad promedio de 150 MB. Las cintas de video tienen una capacidad de almacenado que se mide en gigabytes, sin embrago todavía es muy raro encontrar sistemas que las utilicen. Los discos gusano, son todavía menos comunes quizá debido a su costo; este medio es parecido a las memoria EPROM ya que se puede escribir sobre ellos sólo una vez, gracias a esta característica son utilizados para guardar la información obtenida durante auditorias.

Unix brinda al administrador varios comandos para realizar los respaldos de la información. Uno de los comandos para realizar respaldos es *tar*, este comando sirve para guardar y restaurar archivos, generalmente a partir de un disco flexible o de una cinta. La forma en que opera este comando puede ser controlada mediante sus modificadores. Una de las cosas que debe tomarse en consideración respecto al uso de *tar* es si su usan nombres relativos o absolutos. Cuando se especifican nombres relativos se debe estar ubicado en la posición correcta dentro del

árbol de archivos. Este comando será muy común para los usuarios de ftp muchos de los archivos obtenidos mediante este comando para la transferencia de archivos se encontrarán con el formato *tar*.

Otro de los comandos para realizar respaldos es *cpio*. Este comando funciona de forma similar al anterior, pero tiene una sintaxis más sencilla. Además de ser más sencillo de usar *cpio* es más rápido que *tar* e inclusive se guarda información de una forma más eficiente, ya que intentará leer la cinta varias veces en caso de encontrar algún error y no sólo eso sino que evitará las secciones dañadas de la cinta.

Existe otro comando que además de realizar respaldo de la información realiza conversión de los datos, este comando es *dd*. Si no se especifica ningún tipo de conversión *dd* sólo copia los archivos, sin realizar conversión alguna. Gracias a sus propiedades de conversión este comando se usa comúnmente para restaurar o realizar respaldos en un formato ajeno a Unix, además este comando puede ser utilizado para copiar cintas magnéticas de forma rápida y eficiente.

El comando más común para realizar respaldos es *backup*. Su funcionamiento es muy parecido al comando que se encuentra en MS-DOS. Debe hacerse un respaldo total antes de realizar uno parcial. Se deben utilizar dispositivos de caracteres y no dispositivos de bloque para realizar el respaldo. Este comando puede realizar respaldos en múltiples unidades (volúmenes), indicando al usuario cuando debe cambiar de cartucho o disco. Puede estimar el número de discos o cintas que se utilizarán para realizar el respaldo, esto resulta útil ya que los discos flexibles deben ser formateados antes de ejecutar el respaldo. Con excepción de los mini-cartuchos las cintas no necesitan ser formateadas, pero cualquier tipo de cinta necesita ser re-bobinada una vez terminado el respaldo.

A diferencia de los comandos anteriores *backup* necesita de otro comando para restaurar la información de los discos o cintas hacia la unidad origen. Se habla de restaurar la información ya que todos los comandos que se utilizan para respaldar la información efectúan la compresión de datos para ahorrar espacio. El comando que restaura la información respaldada por *backup* es *restore*. Ambos comandos, *backup* y *restore* también se encuentran presentes en MS-DOS y su funcionamiento es similar.

4.3.6 Periféricos

Para que la operación de un sistema Unix sea completa debe contemplar el manejo de varios periféricos, tal como impresoras, modems, terminales, etc. La instalación, configuración y mantenimiento de estos dispositivos periféricos debe realizarla el administrador del sistema.

La principal diferencia entre los sistemas personales y los sistemas Unix, en lo que a impresión se refiere, es la forma en que se maneja la información a imprimir. En los sistemas personales la información se envía directamente hacia la impresora para ser escrita, por el contrario en los sistemas Unix la información se envía a un buffer conocido como *spooler*. Unix es capaz de transferir varias peticiones de impresión hacia el *spooler* simultáneamente. La impresión del contenido del *spooler* se ejecuta de forma secuencial.

El comando para imprimir en Unix es *lp*, el cual realiza una petición de impresión mediante un número de identificación. El comando además debe determinar el destino de esta petición de impresión, el destino puede ser el nombre de una impresora conectada al sistema o bien el nombre de una clase, que en realidad es un grupo de impresoras.

Cuando se tienen problemas el administrador puede utilizar el comando *cancel* para

suspender los trabajos de impresión pendientes. Otro comando útil para realizar una impresión en Unix es *lpstat*, el cual muestra el estado de los trabajos de impresión, información desplegada por este comando incluye el listado de las peticiones de impresión y los nombres de las impresoras y clases de impresoras.

Para el trabajo de la administración de impresoras se cuenta con el comando *lpadmin*. Este comando configura el servicio de impresión, permite describir impresoras y dispositivos. Se utiliza para añadir o cambiar las impresoras en el sistema, para cambiar el destino por defecto de los trabajos de impresión, para definir las impresoras disponibles para los servicios de impresión remota y para remover impresoras del sistema. La instalación y configuración de una impresora es uno de los procesos que mayormente dependen del sistema en particular de trabajo, en este tipo de tareas es donde programas como *sam* son sumamente útiles.

Para la instalación de otros periféricos Unix cuenta con una herramienta poderosa, el comando *mkdev*. El comando *mkdev* llama a los archivos específicos para la instalación de los dispositivos periféricos. La mayoría de estos archivos de instalación son confidenciales, aunque por lo general no existe necesidad de leerlos. Antes de intentar instalar un dispositivo se debe realizar un respaldo del sistema de archivos y leer las notas de instalación pertinentes. El comando *mkdev* es de uso interactivo, solicita cierta información al administrador para crear el archivo para el manejo del dispositivo asociado con un dispositivo periférico en particular. Cada dispositivo tiene asociado un *mkdev* especial. El comando *mkdev fd* crea un sistema de archivos en un disco flexible. Otra variante a *mkdev* es añadirle *fs* para ejecutar las tareas de mantenimiento necesarias para añadir un nuevo sistema de archivos después que el comando *mknod* creo el dispositivo y que *mkfs* creo el sistema de archivos. Este comando se usa en conjunción con el comando *mkdev*

hd cuando se añade un segundo disco duro al sistema o con el comando *mkdev fd* cuando se crea un sistema de archivos montable en un disco flexible.

El comando *mkdev hd* se usa para crear archivos para el control de dispositivos con un disco duro externo. Como los archivos para el control del disco interno deben coexistir con los del disco externo, *mkdev hd* incluye una sintaxis extendida para el manejo de múltiples controladores. Como su nombre lo indica *mkdev mouse* se usa para generar los archivos para el manejo del mouse. Para dispositivos seriales, se cuenta con el comando *mkdev serial*. Los controladores de los puertos uno y dos ya existen de fabrica, se usa este comando para crear los archivos necesarios en caso de que el sistema sufra expansiones.

Para establecer un sistema multi-usuario se cuenta con el comando para manejo de dispositivos conocido como *mkdev shl*. Este comando configura los parámetros del kernel para establecer el número de sesiones posibles. De igual forma las funciones de comunicación cuentan con su propio controlador, *mkdev streams*. Otro de los controladores comunes es *mkdev tape* que como es claro sirve para el control de las cintas.

Una vez que se ha configurado el manejador, el sistema solicita al usuario permiso para re_enlazar el kernel. Se crean los archivos controladores apropiados en el directorio */dev* y los archivos de configuración del comando *init* solicitan la información necesaria para incluir la configuración de estos dispositivos cada vez que se arranque la máquina. Como el kernel sufrirá modificaciones, esto puede generar errores en el sistema por lo que es conveniente realizar un respaldo de la información antes de tratar de instalar un periférico.

Las terminales tienen una forma de instalación especial. En el archivo */etc/inittab* se describen las terminales instaladas, cada una de las terminales tiene que tener asociado un proceso

getty. El proceso *getty* es el único proceso activo en la terminal, hasta que se utilizada por algún usuario, su objetivo es solicitar el *login* y la contraseña al usuario para que pueda establecer una sesión en la terminal. Además del archivo *inittab* Unix utiliza los archivos *gettydefs* y *gettytab* para la configuración de las terminales.

4.3.7 Acceso y manejo de redes

En la actualidad Unix brinda una cantidad considerable de comandos que permiten a los usuarios accesar servicios y archivos en máquinas remotas. El mantenimiento y configuración de estas herramientas es labor del administrador del sistema Unix. El administrador, o quizá deberíamos decir el gestor, de una máquina Unix es en realidad un asistente del administrador de la red. Estas funciones no están del todo definidas, en ocasiones el administrador de las diferentes máquinas de la red es a la vez el administrador de la red misma. En la introducción del capítulo establecimos una cierta diferencia entre las labores del gestor y las de un administrador de red. Lo establecido en esta sección deberá entenderse como válido para la gestión de redes, es la configuración de una máquina para trabajar en red, no la configuración de la red en si misma.

Las redes Unix se comunican básicamente por medio del protocolo TCP/IP, por lo tanto se identifican dentro de la red por medio de direcciones IP. Además de la direcciones, las máquinas pueden tener un nombre, como se verá en el capítulo siguiente. Por lo regular una máquina tendrá que conectarse más frecuentemente a las máquinas dentro de la red local que con las máquinas fuera de ella, para facilitar la conexión con algunas máquinas en el archivo */etc/hosts* se guarda información acerca de los equipos (hosts) con los cuales se trabaja más frecuentemente. Este archivo contiene la dirección, el nombre y en algunos casos alias de las máquinas importantes

para determinado host. Por ejemplo, la máquina Apollo de la red de la Facultad de Estudios Superiores Cuautitlán contiene en su archivo */etc/hosts* la dirección y nombre de las máquinas Sun que también forman parte de la red de la FES-Cuautitlán, de esta forma cuando deseamos a partir de la estación de trabajo HP Apollo, establecer contacto con la máquina Sun (132.248.102.71), podemos utilizar el alías de la máquina Sun (fescunam) en lugar de utilizar su dirección. En el archivo */etc/hosts* de la máquina Apollo se encuentra la información respecto a la otra máquina.

Otros archivos importantes para la configuración del sistema de red son: el archivo */etc/resolv.conf* y el */etc/*services. El primero contiene las direcciones de las máquinas que funcionarán como servidores de nombres. El concepto de servidor de nombres será explicado en el capítulo seis y el segundo contiene una lista de los servicios disponibles en la red

Los procesos demonio relacionados con la conexión en red entre máquinas mediante el protocolo TCP/IP son los siguientes:

inetd .- La localización de este proceso demonio es */etc/inetd* y su función es monitorear la mayor parte de las operaciones en la red. Este proceso está en ejecución constante y controla a los demás procesos demonio de acuerdo a las instrucciones contenidas en el archivo */etc/inetd.conf*. En los sistemas BSD todos los incisos dentro de este archivo contienen el nombre del servicio, el tipo de socket, el protocolo, el nombre del usuario del programa despachador, el nombre absoluto del programa despachador, etc. Cuando *inetd* recibe un mensaje, lo analiza y después crea una copia del demonio requerido para manejar la petición.

comsat .- El proceso demonio */etc/comsat* notifica a los usuarios que han recibido correo, esto lo hace cuando el usuario entra al sistema.

ftpd .- Para manejar el protocolo de transferencia de archivos (ftp) se utiliza el proceso demonio */etc/ftpd*. El servicio ftp será discutido en el capítulo seis, por ahora basta con decir que este proceso se encarga de la validación de la contraseña del usuario o en el caso de un ftp anónimo se encarga de realiza una copia de los archivos fuera del directorio ftp para aumentar la seguridad del sistema.

gated .- El proceso */etc/gated* se encarga de reemplazar al proceso */etc/routed* , la ventaja del *gated* sobre *routed* está en su capacidad de procesar múltiples protocolos de ruteo y convertir la información de un protocolo a otro.

talkd .- Este proceso se encarga de aceptar las peticiones de conexión realizadas por el comando *talk* e inicia un enlace de red cuando es necesario.

Para realizar las tareas de administración de la configuración de red de una máquina el administrador cuenta con una serie de comandos que le permiten conocer el estado de las comunicaciones en red. Dentro de estos comandos tenemos *rusers*, el cual muestra una lista de todos los usuarios remotos conectados al sistema. Otro comando útil para determinar la calidad de la conexión de un sistema con otro es *ping*. Este comando es una herramienta para la detección de errores en la red. Envía mensajes a una máquina en específico y después brinda información acerca del número de mensajes recibidos.

Para observar el estado de la red el administrador puede utilizar el comando *nstat*, el estado de la red se presenta desde el punto de vista de la máquina que ejecuta el comando. Este comando tiene un gran número de banderas para filtrar la información, a continuación describiremos algunos de los más importantes:

-a Muestra las conexiones Internet activas y su estado.

-i El resultado de este modificador es un resumen de cada interfase de la red.

-m Sirve para mostrar el estado del buffer de la memoria (mbuffer).

-s Este modificador muestra el resumen de los paquetes de cada protocolo.

-r Para ver las tablas de direccionamiento en la red, su estado y un resumen de su uso.

Las máquinas SUN tienen una utileria extra, el programa *traffic*, el cual muestra de forma gráfica la carga de la red, el tamaño de los paquetes, el protocolo utilizado y el origen y destino de los paquetes. Para ejecutar este programa se necesita la ejecución de otros procesos demonio, consulte el manual para más detalles. SUN además soporta un herramienta llamada *etherfind* que puede ser usada para rastrear paquetes con determinadas caracteristicas, de la misma forma en que lo hacen los analizadores de red.

4.4 Seguridad en un sistema Unix

La seguridad de un sistema es de suma importancia, sobre todo en la actualidad, ya que cada uno de los sistemas en los que trabajamos pueden ser la puerta hacia una cantidad enorme de máquinas. Debido a la importancia de la seguridad en los sistemas, se le asigna un nivel completo a este aspecto de la administración en el modelo de administración OSI.

Como mencionamos en la introducción de este capítulo, Unix fue creado con la intención de compartir información por lo que, en su versión original, no incluye ninguna medida de seguridad mas que el uso de las claves de acceso y los permisos de los archivos.

Debido a esto ningún sistema Unix podrá considerarse como completamente protegido, cualquier persona, con la suficiente paciencia y con buenos conocimientos respecto de este

sistema operativo puede violar hasta la máquina mejor protegida. Como administradores debemos hacer lo más difícil posible la entrada no autorizada al sistema y establecer medidas de seguridad para evitar daños debido a acceso no autorizado.

Existen dos políticas para establecer seguridad, tratar de evitar la entrada no autorizada al sistema, tapando todos los huecos que tiene Unix en lo referente a seguridad. O bien asumir que el acceso no autorizado ocurrirá de cualquier forma, y por lo tanto establecer medidas encaminadas a evitar que un usuario ajeno al sistema pueda causarle algún daño. En el presente capítulo nos enfocaremos a señalar los puntos más débiles de Unix, en la mayoría de los casos errores de administración de la máquina dejan al sistema a merced de los intrusos e inclusive los intrusos más novatos pueden accesar a la máquina con relativa facilidad. Los primeros ataques resultaban en general inofensivos, sin embargo ahora los ataques tienden a dañar la información, por lo tanto una buena política de respaldos puede resultar una buena ayuda para minimizar los efectos de ataques maliciosos.

En general Unix tiene problemas para garantizar la seguridad de un sistema, algunos de estos problemas siguen sin ser solucionados y otros sólo se han solucionado en algunas versiones de Unix y en otras no. Además, no todas las máquinas trabajan con la última versión del sistema operativo, ya sea por su costo, disponibilidad, etc. Por lo tanto aunque el fabricante solucione algún problema, los usuarios no podrán obtener los beneficios de este arreglo a menos que adquieran la nueva versión del sistema; de donde, no resulta conveniente ni ético señalar la forma en que se puede violar la privacidad de un sistema, no deseamos dar un recetario de como se puede tener acceso no autorizado a un sistema, por el contrario mencionaremos algunas formas de aumentar la seguridad de un sistema sin profundizar mucho en la forma en que el sistema

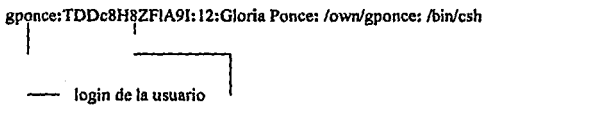
puede ser violado. Esta forma de plantear los temas de seguridad es la utilizada por la Universidad de Berkley, en las mejoras a su versión de Unix (BSD Unix), se establece la forma de implementar la mejora sin discutir en gran detalle el problema. Como administrador de un sistema se debe estar en busca de medidas para aumentar la seguridad de un sistema y se debe ser discreto con los problemas de seguridad que puedan detectarse. La mejor fuente para obtener información acerca de problemas de seguridad y de como resolverlos es de otros administradores, esta es la función de los grupos de discusión como el establecido por la Universidad de Berkley "Fixes security hole".

4.4.1 Problemas con las contraseñas (passwords)

El problema más común para comprometer la seguridad de un sistema es la elección de claves de acceso triviales, este problema se hizo manifiesto durante el otoño de 1988 durante el cual un programa conocido como el Gusano de Internet, el cual uso un algoritmo muy sencillo para adivinar las claves de acceso y dos fallas en Unix, una en el programa Mail y otra en una de las bibliotecas de "C", para infectar cientos de máquinas a lo largo de los Estados Unidos en cuestión de horas. Pero su principal forma de ataque fue mediante el uso de una lista de 400 contraseñas más populares y aprovechando las facilidades de ejecución remota de comandos dadas por Unix.

Aunque este programa sólo pudo atacar dos arquitecturas en particular, pudo accesar a diferentes sitios, Universidades, algunas cedes Gubernamentales y organizaciones comerciales. La rapidez con la que se extendió este programa a lo largo de Internet puso de manifiesto la necesidad de medidas de seguridad más estrictas.

Otro de los problemas respecto a las claves de acceso es la existencia de los mismos, como mencionamos en un apartado anterior existen usuarios reales y usuarios lógicos, los procesos demonio son un ejemplo de seudo-usuarios o usuarios lógicos. En algunos sistemas los usuarios lógicos, no tienen asignada una contraseña, lo que deja al sistema a merced de los intrusos. El administrador debe asegurarse que estos procesos tengan asignada una contraseña, para conocer el estado de las contraseñas de estos usuarios lógicos se debe consultar el archivo */etc/password*. En este archivo se almacenan las claves de acceso de todos los usuarios, tanto lógicos como físicos. Las contraseñas se encuentran cifradas, para evitar que alguien pueda obtenerlas con la lectura de este archivo. Aquí se muestra un ejemplo de una línea del archivo */etc/password*, los diferentes campos se separan mediante los dos puntos (:).



Los seudo-usuarios, como lo son los procesos demonio, deben tener un * en el campo de la clave de acceso, estos seudo-usuarios son utilizados, pero en realidad nunca accesan al sistema. Con el asterisco en el campo de la contraseña se le indica al sistema que no existe ninguna contraseña válida para con la cual estos procesos puedan accesar al sistema. Adicionalmente, como las claves de acceso son manejadas por los usuarios directamente, estos pueden retirar su clave de acceso, declarando una clave nula (dando acceso a cualquiera, ya que los ids de

presentación son públicos). El administrador debe revisar el archivo */etc/password* regularmente para evitar que existan cuentas con claves nulas. Existen muchos programas que se encargan de revisar el archivo */etc/password* para detectar problemas de seguridad, pero el comando *awk -F '{if($2==“”)print $1}' /etc/passwd* es bastante bueno para encontrar cuantas son claves nulas.

Como el administrador del sistema (root) tiene el completo control de la máquina su clave de acceso debe ser la más segura de todas, debe cambiarse periódicamente, se recomienda que sea una clave fácil de teclear para evitar que una persona mirando por encima del hombro pueda descubrirla, debe evitarse el uso de claves significativas, pero sobre todo debe ser privada; ninguna de las precauciones anteriores tiene sentido si la clave del administrador es pública. Esto puede parecer obvio; sin embargo, es asombrosamente frecuente encontrar instituciones en la cuales la contraseña del administrador es conocida por todos los usuarios. En casos extremos la clave de acceso se encuentra visible cerca de la máquina (por ejemplo, en un papel pegado a un lado del monitor) para que cualquiera pueda hacer uso de ella. Debe estarse consciente que no hay razón para que todos los usuarios deben conocer la contraseña del super-usuario, si todos tienen, privilegios de administración sobre el sistema, este terminará siendo incontrolable.

4.4.2 Problemas con el identificador del usuario (User ID)

Dentro del sistema cada usuario es identificado por un número, el UID. El super-usuario tiene un UID especial, cero, este número especial permite que ciertas operaciones sólo puedan ser ejecutadas por el administrador. El UID también se encuentra almacenado en el archivo */etc/password*, el problema radica en que en este archivo puede haber más de un usuario al cual se le asigne el UID cero. De esta manera, cuando un intruso logra obtener un shell de super-usuario

(por algún medio), trata de establecer una cuenta en */etc/password* que le permita accesar al sistema con derechos de super-usuario, pero usando un login diferente para evitar ser descubierto mediante el comando *who*. El comando *who* da una lista de los usuarios en el sistema mediante la lectura del archivo */etc/utmp* el cual no contiene el UID de los usuarios, por lo tanto no puede detectar que alguien dentro de la red tiene derechos de administración. Para defenderse de este tipo de ataques, se puede usar un comando similar al utilizado para detectar cuentas con contraseña nula, *awk -F '{if ($3==0)print $1}' /etc/passwd*. Este comando también puede ser adaptado para encontrar identificadores de grupos o usuarios con marcas raras, por ejemplo una cuenta en */etc/passwd* sin nombre de usuario o con un signo de puntuación como nombre puede parecer sin sentido; pero, puede ser usada para accesar al sistema.

Otro problema con el UID, surge con los programas capaces de cambiarlo. Por ejemplo, cada usuario tiene derecho a manejar su contraseña, pero sólo el administrador tiene derecho a escribir sobre el archivo */etc/passwd*, para que el usuario normal pueda modificar su contraseña debe hacerlo como super-usuario. Por lo tanto el programa *passwd* cambia el UID del usuario a cero para que pueda modificar su contraseña en el archivo */etc/passwd*. Programas como *passwd* no presentan problema, ya que su operación es local, mediante este comando sólo se puede operar en el archivo */etc/passwd*. Sin embargo, un programa editor del super-usuario que sea capaz de modificar el UID permitiría a cualquier usuario modificar el archivo */etc/passwd* o cualquier otro archivo. Aprovechando este defecto en Unix, un usuario mal intencionado puede realizar un ataque sushi. Este ataque consiste en obtener la identidad del super-usuario, para copiar el *shell* del adm inistrador a un archivo, "sushi" por ejemplo, una vez hecho esto se cambian los permisos de sushi y el usuario puede regresar a su identidad original. La copia del *shell* creada es un

programa capaz de cambiar el UID y el dueño de este archivo es el administrador, de esta manera con la simple ejecución del archivo "sushi" permite al usuario normal obtener todos los derechos sobre el sistema.

Para evitar este tipo de ataque: no se debe permitir a nadie usar el shell del super-usuario; sólo el administrador debe tener derecho a escribir programas capaces de modificar el UID de los archivos; no confiar en que el archivo se llamará sushi, el autor del ataque puede cambiar el *shell* robado a un archivo que tenga un nombre más difícil de identificar, para detectar este tipo de ataques se puede ejecutar el siguiente programa:

```
find / -user root -perm -400 -exec ls -l {} \;
        | mail root  # setuid
find `echo $PATH | tr ":" " " ` -perm -002 -exec ls -l {} \;
        | mail root # writable
```

4.4.3 Revisión del archivo /usr/spool/cron/crontab

Los archivos del directorio /usr/spool/cron/crontabs (así como algunas de las primeras versiones de /usr/lib/cron/crontabs y /usr/lib/crontab) son muy útiles, pero deben ser usados de forma inteligente o de lo contrario pueden ser usados para penetrar al sistema. Sólo el administrador debe tener derecho a leer los archivos de este tipo dedicados a la administración, de lo contrario se corre el riesgo de que un intruso pueda crear un programa capaz de leer los modos de ejecución de los programas en cada *cron* para después hacer una copia de la parte superior de

los programas con permiso de escritura, como la mayoría de los *cron* ejecutan las instrucciones como super-usuario el hacer esto le permitiría al intruso atacar el sistema fácilmente.

El comando *at* presenta un problema aún mayor, este comando permite al usuario normal ejecutar comandos en un determinado horario. Para poder hacer esto, se ejecuta cada diez minutos el proceso demonio llamado /usr/lib/aturn en el *cron*. Cuando algún usuario ejecuta el comando *at*, se crea un archivo en el directorio /usr/spool/at, el cual contendrá los comandos a ejecutar. Como este proceso demonio debe ser capaz de ejecutar cualquier programa es del tipo señalado en el apartado anterior, un programa perteneciente al administrador y capaz de modificar el UID. Esto deja al ofensor varias alternativas:

a) Ejecutar los comandos necesarios para obtener un programa "sushi" mediante el comando at.

b) Ejecutar un programa inofensivo mediante at, pero editar el archivo creado dentro del subdirectorio /usr/spool/at para añadir el código de creación de un programa sushi.

c) Hacer una copia del archivo en /usr/spool/at, añadir el código para la creación de un programa sushi a la copia y renombrar la copia de manera que aturn la ejecute en un momento determinado.

La opción a tomar dependerá de los huecos que se encuentren todavia descubiertos, en algunas ocasiones basta cambiar la pertenencia del archivo en /usr/spool/at, asignándoselo al administrador, para engañar al comando at. Para prevenirse en contra de este tipo de ataque asegúrese que el directorio /usr/spool/at pertenece al administrador y que tiene asignados todos los permisos para el dueño (el super-usuario), y que el resto de los usuarios, incluyendo a los miembros del grupo, sólo tengan permiso de lectura y ejecución. Es conveniente probar las

técnicas antes descritas, si aún con los cambios en /usr/spool/at se logra establecer un programa sushi por alguno de estos métodos, no quedará más opción que desactivar el proceso demonio *aturn* de lo contrario el riesgo de ser víctima de un ataque de este tipo seguirá latente.

4.4.4 Permisos de archivos importantes

Existen varios archivos que deben tener permisos de forma particular para evitar problema de seguridad. Algunos vendedores configuran sus equipos con permisos del sistema de archivos para una máquina que trabajará en un ambiente amigable y de cooperación; sin embargo, esto no resulta siempre así. A continuación mencionaremos algunos de los archivos más conocidos y los permisos que deben tener:

El archivo /usr/lib/L.sys contiene los nombres y contraseñas de todos los vecinos uucp de un sistema Unix. Si este archivo tiene permisos incorrectos, puede ser utilizado para violar la seguridad de los sitios donde se encuentran los vecinos. El archivo L.sys debe tener permisos de lectura y escritura para el dueño (uucp), sólo permiso de lectura para el grupo (daemon)y ningún permiso para el resto. El contenido de este archivo debe ser revisado para asegurarse que sólo contiene información acerca de los sitios a los que realmente se desea conectar.

El archivo especial /dev/kmem permite el acceso al kernel, es usado por comandos como *ps* que necesitan conocer la estructura de los datos en el kernel. Este archivo sólo debe ser leída por el dueño y los miembros del grupo. Algunas versiones de Unix permiten que este archivo pueda ser leído de forma pública, lo que ocasiona un problema de seguridad mayor, ya que un programador lo suficientemente capaz podría tener acceso a información como contraseñas descifradas leyendo las estructuras de datos y los buffers del kernel. Si el archivo /dev/kmem

puede ser leído por cualquiera, cambie los permisos inmediatamente, si al hacer esto algunos programas presentan problemas para ser ejecutados, cambie su identificador de grupo, asignandoles el identificador de grupo que tenga kmem.

Aunque puede resultar de cierta forma obvio, los archivos /etc/password y /etc/group sólo deben poder ser modificados por el administrador. Ambos deben pertenecer al super-usuario y deben tener permisos de lectura y escritura para el dueño (root), y sólo de permisos de lectura tanto para los miembros del grupo (debe pertenecer a algún grupo del sistema, daemon por ejemplo) y para el resto de los usuarios.

Otra fuente potencial de problemas son los archivos de dispositivos, tales como las particiones de disco duro. El tener permiso de lectura y escritura sobre un dispositivo de disco equivale a tener los mismos derechos sobre el sistema de archivos que contiene. Solamente el administrador debe tener ambos permisos, el grupo algunas veces tiene permisos de lectura; pero el resto de los usuarios no debe tener ningún permiso sobre este tipo de archivos.

4.4.5 Problemas con las terminales inteligentes

Muchas terminales pueden operar de forma independiente de la computadora huésped, aunque Unix no aprovecha esto de forma directa algunas terminales de este tipo pueden generar problemas. En particular, se pueden enviar señales de control hacia la mayoría de estas terminales para hacer que envíen un eco de lo que se les envío. Si esta característica es usada para enviar eco de comandos Unix, los comando pueden ser ejecutados como si fueran enviados por la persona trabajando en la terminal. De esta forma si el super-usuario es quien está trabajando en la terminal, la seguridad del sistema completo se encuentra en peligro.

Este problema es conocido como el defecto de la línea 25 (25th line bug) porque en la mayoría de las terminales inteligentes los caracteres que se envían como eco se encuentran en la línea 25. Una defensa en contra de ataques de este tipo es usar el comando *mesg n*, el cual inhibe cualquier señal enviada directamente a la terminal. La desventaja de este comando, es que se inhibe el funcionamiento de programas como *write* o *talk*. Para evitar este ataque sin tener este problema se puede hacer que la terminal emule a otra terminal en la cual no soporte la línea 25.

Capítulo V

Seguridad

El ejemplo más claro de la utilidad de las redes es la red mundial conocida como Internet. Esta red crece de forma exponencial, paso de un poco más de 100,000 nodos (hace 6 años aproximadamente) a más de 2,500,00 en la actualidad; pero, Internet no sólo ha sufrido cambios en la cantidad de usuarios conectados a ella, la red misma está cambiando. En un principio se encargaba de la comunicación entre las grandes Universidades y Centros de Investigación alrededor del mundo, sus intereses principales eran apoyar a la educación y promover la investigación, involucrando a los estudiantes con los problemas tecnológicos actuales. En nuestros días, el uso de esta red se ha ampliado para incluir propósitos comerciales. Tanto la parte comercial como la académica han influido en el desarrollo de Internet, la consecuencia principal del crecimiento de esta red, ha sido la increíble cantidad de información que se puede obtener a través de ella. Dentro de esta información tenemos resultados de investigaciones, transacciones financieras, información privada de diferentes compañías e información personal financiera. El valor de la información a través de Internet es tan valiosa que algunas personas desean obtenerla aún usando medios no autorizados. En este capítulo estudiaremos uno de los más grandes retos de los administradores de Internet y de cualquier administrador en general, garantizar la privacía e integridad de la información dentro de la red. Nos enfocamos en Internet ya que al ser una red de gran escala las soluciones construidas en esta pueden ser casos particulares de las redes más pequeñas.

En el capítulo dedicado a la administración de Unix mencionamos que no existe ningún sistema de seguridad infalible y cuando se trata de una red de la magnitud de Internet, esto resulta aún menos factible, entre más se comparte mayor es el riesgo de un acceso no autorizado. La única forma de tener un sistema completamente seguro, es tenerlo fuera de la red y con vigilancia estricta las veinticuatro horas del día. A esto hay que añadir que Internet, como Unix, fue creada con el propósito de compartir información, ninguno de los dos fue creado tomando la seguridad como prioridad[1]. Los problemas de seguridad cobran importancia a partir del ataque del gusano de Internet, antes del ataque de este programa la mayoría de los administradores conocía los problemas de seguridad en Unix, sabían que en teoría un ataque de este tipo podría ser realizado; sin embargo, la mayoría de los administradores se sorprendieron al ver el ataque convertido en realidad. La moraleja del ataque del gusano fue demostrar que los ataques a la red son reales, no sólo especulaciones teóricas.

Pero no sólo el ataque del gusano forzó a los administradores a aumentar las medidas de seguridad. De la misma forma que Unix, Internet sufrió un cambio en el ambiente de su aplicación. Al incluir Internet propósitos comerciales tiene que buscarse la forma de garantizar no sólo por la privacidad de la información sino también el proteger los derechos de autor de las obras contenidas en las máquinas de la red y de las transmitidas a través de ella.

A todo esto hay que añadir la alarma generada por la nueva forma de ataque a las redes, el monitoreo no autorizado del tráfico de una red, también conocido como inspección de paquetes (packet sniffing).

---

[1] Multics, el sistema a partir del cual fue generado Unix, tenía como prioridad la seguridad y en la actualidad es uno de los sistemas operativos más seguros. Unix al pretender ser una simplificación de MULTICS su principal objetivo era establecer un medio para compartir información de forma sencilla.

En febrero del año pasado, sucedió algo que llamó la atención del Centro de Control de Internet (CERT). En corto período de tiempo, el número de reportes de inspectores clandestinos (sniffers) aumento considerablemente. La importancia en este tipo de ataque es que en realidad es muy sofisticado, es difícil de detectar; pero, no sólo eso sino que el aumento en el número de casos detectados nos habla que además se encuentra ampliamente difundido. Esto permite que intrusos novatos con deseos de atacar una red puedan hacerlo aún sin contar con los conocimientos necesarios para llevarlos a cabo. El CERT ha reportado casos de sistemas penetrados mediante formas de ataque bastante sofisticadas, en los cuales sólo se ha podido detectar el ataque porque el intruso, demostrando su ignorancia, trata de ejecutar comandos del sistema operativo MS-DOS.

Es necesario entender que la lucha por tener un sistema seguro es constante, conforme se descubren formas de evitar alguna forma de ataque se generan nuevas. Para aumentar la seguridad de nuestros sistemas es conveniente estar en contacto, formar grupos de trabajo (constituidos por administradores) para intercambiar información y experiencias así como difundir medidas de seguridad, de la misma forma en que los intrusos la intercambian medios para realizar los ataques. La guerra apenas comienza y conforme aumenta y se populariza el uso de la red cada batalla cobrará importancia.

5.1 Seguridad de un sistema de cómputo

La seguridad de un sistema de cómputo, una red por ejemplo, no consiste en guardias y sistemas de alarma o detectores infrarrojos en las entradas de acceso a los edificios donde se encuentran estos sistemas. Este tipo de protección estaría enfocada a evitar robos, más que a proteger el sistema en si mismo. La seguridad de un sistema se refiere a la protección de la información dentro de la computadora y del control de uso de los recursos del mismo. Se debe proteger al sistema en contra de accesos no autorizados, de vigilar a los usuarios para saber que comandos y programas ejecutan, evitando así que alguien pueda abusar de los recursos del sistema, y quizá lo más importante, proteger la información. Podemos dividir estas funciones en dos grandes categorías: medidas para evitar el acceso no autorizado, los muros contra incendio (Fire Walls) y el programa npasswd son medidas de este tipo; y establecer formas de validar los servicios que ofrece la red a los usuarios, kerberos es un ejemplo de es tipo de sistema. Para tener medidas de seguridad funcionales se deben cubrir básicamente tres aspectos:

- Tener una política de registro: Esto ayudará al administrador a conocer que comandos se han ejecutado y quien se encargo de realizarlos, se deberán registrar todas las actividades de los usuarios, identificando a los ejecutores de cada las actividades. Los registros pueden ser revisados para encontrar actividades sospechosas o para determinar si alguien a violado la seguridad del sistema. Cada usuario debe ser identificado de forma única al momento de iniciar su sesión en el sistema, de esta forma se podrán asignar los registros correspondientes, la propiedad de los archivos y directorios, y se podrá determinar los derechos de acceso a los diferentes archivos. Esto lo hace Unix de forma automática, e inclusive tiene integrados sistemas para llevar la contabilidad de los recursos del sistema usados por los diferentes usuarios.

- Protección del sistema: Prevenir el uso no autorizado de los recursos del sistema. Los recursos a defender dependerán de cada sistema en particular, máquinas como la Cray en las cuales los ciclos de operación del CPU son muy costos, deberá tener medidas de seguridad encaminadas a evitar el uso de este recurso de forma indiscriminada, pero en sistemas como estaciones de trabajo o computadoras personales es más conveniente proteger la información que contienen, el espacio en el disco duro, etc. Las medidas encaminadas a la protección de los recursos refuerzan al siguiente punto, la protección de la información. Por ejemplo el acceso al CPU de una máquina puede brindar al intruso la oportunidad de conocer información crítica del sistema como las claves de acceso, ya que en el CPU estas se encuentran no cifradas.

- Protección de la información: Control sobre el acceso a la información, se debe cuidar los derechos de los diferentes usuarios a leerla, copiarla o modificarla. Es necesario tomar medidas de seguridad para prevenir tanto ataques externos como de ataques internos. Un buen sistema de respaldos de la información resultará sumamente conveniente.

Queremos resaltar que al estar conectada a Internet una máquina no sólo debe preocuparse por su seguridad, sino que debe evitar ser un puente para atacar otro sistema. Mucha gente piensa que la información en su sistema no es tan importante y por lo tanto no se preocupan de establecer ninguna política de seguridad, olvidándose que mediante su sistema se puede comprometer la seguridad de otros equipos.

5.2 Políticas de seguridad

Para hacer un sistema seguro es necesario establecer toda una política de seguridad, en esta política debe tomarse en cuenta el costo de establecer las medidas de seguridad y la forma en que el funcionamiento del sistema se verá afectado por el establecimiento de estas medidas de seguridad. No se debe olvidar que la popularidad de Unix, y por la tanto de las redes que usan este sistema operativo, se debe en gran medida a las facilidades brindadas para intercambiar información entre sistemas completamente diferentes. Al aumentar la seguridad de un sistema se reduce la facilidad de transmitir información entre máquinas, por lo tanto si se añaden medidas de seguridad sin una motivación seria, los usuarios trataran de saltar estas medidas y si la seguridad se aumenta de tal forma que el sistema se vuelve muy difícil de manejar, ningún usuario estará dispuesto a usarlo. Por lo tanto, la facilidad de uso del sistema debe balancearse con las medidas de seguridad necesarias.

Por lo regular el punto más débil de una red son los usuarios, en el capítulo dedicado a la administración de Unix mencionamos algunos problemas referentes a la creación de claves de acceso, pero eso no es todo, hasta el sistema de contraseñas y ciframeinto de información mejor diseñado, será inútil si los usuarios olvidan cerrar sus sesiones al abandonar el sistema; situación que es muy general. El establecer una política de seguridad ayuda a concientizar a los usuarios de las cosas que están permitidas y de las que están prohibidas, es conveniente establecer por escrito las políticas de uso de la red. Este documento debe ser redactado por el administrador y firmado por cada uno de los usuarios. Un sistema de este tipo es usado en la UNAM para otorgar claves de acceso para la CRAY. Al aceptar la solicitud de un usuario la DGSCA entrega a éste un documento donde se encuentra su clave de identificación (login) y su contraseña (password).

En este documento se incluye un texto en el cual se describe usuario la política de uso de esta máquina. El usuario al recibir su clave debe firmar un documento donde consta que tiene conocimiento de esta política.

La política de seguridad debe dejar claro que está prohibido usar el acceso a la red con fines ajenos a la empresa u organización a la que pertenece (las claves a RedUNAM no pueden usarse para fines comerciales, por ejemplo). Las reglas de conducta deben tener un apartado dedicado al administrador del sistema. Se han dado casos en los cuales el administrador leen la correspondencia de los usuarios; por lo tanto, la política de seguridad también debe establecer los límites del poder del super-usuario, para evitar abusos. Adicionalmente la política de seguridad debe especificar las penalidades a las que se harán acreedores los usuarios que violen las reglas establecidas por esta política.

La política de seguridad no sólo actúa como freno sino que hace más fácil la persecución legal, porque las reglas de uso correcto del sistema -y las penalidades para castigar un comportamiento no aceptable- están claramente establecidas. Por ejemplo, los sistemas donde se requieren la clave de identificación y la contraseña, debe poner en claro que el acceso al sistema está limitado sólo a los usuarios autorizados. Existió un caso en el cual un intruso fue absuelto de la acusación de haber violado un sistema, solamente porque el mensaje donde se solicitaba la clave de identificación decía "Bienvenido".

En la política de seguridad se debe incluir un plan de respuesta ante un ataque, para hacer más sencillo el responder a un ataque. En general los pasos a seguir para responder al ataque son los siguientes:

- Actuar de acuerdo a la política establecida.

- Verificar el incidente.

- Determinar la magnitud y el alcance de la intromisión

- Comunicar el problema y las acciones tomadas al responsable de la red, a los grupos de emergencia (si existen), a todos los sitios afectados y en caso de ser necesario a una agencia investigadora (como se verá en el caso del intruso astuto "Wily Hacker").

- Restaurar el software del sistema: restaurar programas, aplicaciones desde su fuente original y restaurar los datos a partir de los respaldos (que deben ser periódicos).

- Realizar un reporte, donde se incluya información como horario, recursos utilizados y el daño ocasionado por el ataque. Haga un documento donde explique los problemas, para usarlo como experiencia y actualice las políticas de seguridad según sea necesario.

En el primer punto insistimos en la política de seguridad; desgraciadamente, muchas organizaciones no tienen una política de seguridad y cuando esta existe, no es más que una política de uso adecuado (es decir sólo una parte de lo que es la política de seguridad). Sin la política de respuesta al ataque el administrador (gestor) puede perder tiempo precioso, deliberando si debe desconectar el sistema de la red o no, si tiene o no derecho a hacerlo, con la desventaja adicional de poder comunicarse con el administrador de la red ya que raramente los ataques suceden en horarios de trabajo, por lo regular se realizan a altas horas de la noche. Además, la política de seguridad puede ayudar a intentar perseguir legalmente a los intrusos, tratando de obtener pruebas desde el inicio del ataque.

Como los usuarios forman parte fundamental en la seguridad del sistema, deben estar involucrados con ella para que esta tenga buenos resultados.

Deben estar conscientes que los principales afectados por una intromisión al sistema pueden ser ellos, por lo tanto deben respetar las medidas de seguridad establecidas. Deben estar conscientes tanto de las ventajas de poder obtener software público, como de los riesgos que se corren. Dentro de los riesgos del software público, existen sistemas que aparentan ser inofensivos, pero que en realidad contienen formas de ataque al sistema. Uno de los casos más conocidos de este tipo de problemas fue el generado por el "caballo de Troya" conocido como "Turkey", este software presumía dibujar un pavo en la pantalla de la computadora, pero en realidad lo que hacía era borrar todos los archivos del subdirectorio del usuario. No se debe instalar software del cual no se tenga el código fuente, y aún así se deben extremar precauciones.

Otra responsabilidad de los usuarios es ser respetuosos con sus vecinos en la red. Aunque en realidad no se tenga la intención de causar problemas, ciertas actividades pueden ser interpretadas como intentos de forzar un sistema; por ejemplo, accesar a un sistema por medio de un hoyo en la seguridad, revisar si tiene servicios que no requieren identificación o un sistema de archivos montable. Como este tipo de actividades son indicadores de un ataque, muchos sitios perderían tiempo rastreando la fuente de estas actividades, hasta estar seguro que no se trata de una intromisión.

Por último, la política de seguridad también debe prever ataques internos, recuerde que los trabajadores molestos tienen más razones para desear un daño en el sistema que cualquier intruso exterior. En Estados Unidos se han realizado estadísticas donde se demuestra que la mayor parte de los ataques provienen del interior. Es común el robo de información confidencial, ya sea con el ánimo de dañar o para venderla a compañías competidoras.

5.3 Formas de ataque mejor conocidas

El conocer las formas de ataque que han sido usadas nos brindará la oportunidad de protegernos contra ellas. El objetivo de esta parte es establecer el primer frente de seguridad de un sistema, impedir el acceso no autorizado. Si los intrusos no pueden obtener más que el mensaje de presentación del sistema, la información se encontrará segura. No analizaremos en gran detalle los problemas de Unix que fueron y son explotados para realizar los ataques, ya que no es nuestra intención convertirnos dar un recetario de como romper la seguridad de los sistemas. Deseamos, que mediante la mención de los errores los administradores revisen sus sistemas para asegurarse que los huecos se encuentran debidamente cubiertos. Desgraciadamente esto también puede despertar la curiosidad de los intrusos; sin embargo, confiamos en que las personas que leerán esta Tesis, al ser universitarios, sabrán hacer uso de la información aquí brindada.

En esta sección también demostraremos como las formas de ataque han ido evolucionado, día a día se hacen más complejas y también más maliciosas. Por ejemplo, el primer ataque serio realizado, el del gusano de Internet, sólo pudo infectar ciertos tipos de máquinas (quien halla creado un programa para ser transportado a un sistema diferente o haya intentado compilar alguno de los programas de la red, entenderá el porque) y, en realidad, no causaba ningún daño serio al sistema bajo ataque. Ahora, los ataques son mucho más complicados y perversos, algunos no sólo tienen la intención de robar información sino también de destruirla. Por lo tanto debemos insistir en que para que las medidas de seguridad puedan resultar realmente exitosas se debe contar con un buen respaldo de la información.

5.3.1 El gusano de Internet

El ataque más famoso y también uno de los más comentados fue el realizado el dos de noviembre de 1988, el llamado gusano de Internet infecto, en el curso de una noche, cientos de máquinas a través de toda la Internet. El gusano explotó tres defectos de la versión de Unix creada por la Universidad de Berkley (BSD Unix), estos defectos permitían a un usuario accesar a una máquina saltando las medidas de seguridad. Usando estos defectos la máquina ofensora envía un programa que tiene como objetivo compilar y ejecutar otro programa pequeño, para establecer un lazo de comunicación entre la máquina invadida y la ofensora. Una vez garantizado el enlace, se envía una versión pre-compilada de programa que se encarga de la creación del gusano, si el programa de creación se lograba compilar y ejecutar de forma exitosa, la máquina víctima se convertía en una nueva ofensora e iniciaba el ataque hacia otras máquinas.

Uno de los defectos de Unix, utilizado por el gusano fue la función `debug()` del programa *sendmail*. Entre otras características, el programa *sendmail* incluía la habilidad de enviar correo a un programa, de forma que éste pueda ser ejecutado con el cuerpo del mensaje como entrada. Generalmente esto sólo es permitido cuando el programa es del tipo de *mail* o con archivos con encabezados de un usuario del sistema. De esta manera el usuario puede ejecutar programas aún en periodos de vacaciones, pero en sistemas donde la función `debug()` se encuentra disponible, esta cualidad de *sendmail* se amplia aún a conexiones remotas. En el programa *sendmail* incluido dentro de la versión 4.3 BSD y la anterior a la 4.1 de SunOS tienen disponible la función `debug()`. El gusano utiliza esto, conectándose al proceso demonio *sendmail*, donde el mensaje va dirigido al shell y tiene como propósito compilar el programa que se encargará de empezar el ataque.

Un segundo defecto usado por el gusano, se encuentra en el programa *finger*. Este programa tiene como propósito brindar información acerca de los usuarios del sistema y puede obtener su línea de entrada a partir de una máquina remota, la línea de entrada se usa para especificar la persona acerca de la cual se desea buscar información. Desafortunadamente, la rutina utilizada para leer la línea de entrada no tiene ninguna forma de revisión del rango y por lo tanto no puede prevenir un sobre flujo del buffer. Como el buffer se encuentra en la pila (stack), el sobre flujo permite crear el marco para una nueva pila; con lo que se permite la ejecución de una pequeña pieza de código (proporcionada por el gusano). La pieza de código a ejecutar entonces se encarga de establecer un *shell* que recibe su entrada del sistema remoto. Entonces, sin ninguna dificultad se puede realizar el ataque. Este tipo de ataque depende de la arquitectura del sistema, por lo que funcionó muy bien en las máquinas Vax, pero falló en las Sun.

El tercer defecto utilizado por el gusano era quizá el más conocido por los administradores de sistemas, la función `rexec()` del programa *rsh*. Tanto el programa como la función proporcionan un shell en una máquina remota. La función requiere la identificación del usuario por medio de su contraseña (password), pero el programa *rsh* permite el acceso de ciertos usuarios a algunas máquinas en específico sin pedir identificación, valida los derechos de un usuario para ejecutar comandos de forma remota por medio de la información contenida en los archivos *host.equiv* y los archivos *.rhosts* de cada uno de los usuarios. El gusano primero ataca al sistema usando *rsh* con mismo nombre que el usado en la máquina local. Si esto falla entonces trata de adivinar la contraseña de cualquier cuenta en la máquina local, si logra obtener alguna contraseña ejecuta el comando `rexec()` en todas las máquinas listadas en el archivo *.rhosts*, usando la contraseña descubierta con una lista de todos los nombres de los usuarios del sistema.

Si esto también falla entonces, se conecta a la máquina local con la contraseña recién descubierta y utiliza el programa *rsh*, como si fuera ese usuario, confiando en que tenga derecho a entrar en alguno de los sistemas remotos.

El ataque del gusano se realiza de la siguiente forma: primero intenta un ataque por medio de *rsh* y `rexec()`, si este ataque falla entonces se intenta un ataque por medio de *finger* y finalmente como último recurso el ataque por medio de *sendmail*. El gusano además usa una gran cantidad de trucos para cubrir sus acciones, entre las que se incluyen: borrar su lista de argumentos, borrando sus archivos tan rápido como sea posible, escondiendo su nombre bajo el sobrenombre de *sh* y re-invocandose cada determinado periodo de tiempo de forma que los programas ejecutados sean cortos.

Robert Tappan Morris (creador del gusano de Internet) fue la primera persona en ser condenada por un jurado bajo el cargo de fraude computacional (Computer Fraud), por accesar a una red gubernamental sin tener autorización para hacerlo. La condena impuesta a Morris fue de una fianza de 10,000 dólares, 400 horas de trabajo para servicio de la comunidad y tres años bajo libertad condicional. Los intentos de sus abogados para apelar la sentencia, argumentando que la intención de este ciudadano no era la de lastimar, resultaron infructuosos.

5.3.2 El intruso astuto[2]

El ataque del "intruso astuto" no es un algoritmo de ataque como lo es el gusano, descrito en la sección anterior, en realidad es un caso de ataque a varias redes con el fin de robar información importante.

[2] El término original es "Wily Hacker", no hemos encontrado una traducción más adecuada para el término Hacker que el aquí presntado; según el diccionario, puede también traducirse como cortador o mercenario.

Este caso debe su importancia a la magnitud de los propósitos perseguidos por los autores del ataque. Todo empezó como un simple caso de abuso de los recursos de la red del Laboratory Lawrence de la Universidad de Berkley. Clifford Stoll buscando la causa de una discrepancia de *setenta y cinco centavos de dólar*, en el sistema de contabilidad de la computadora del laboratorio encontró una cuenta sin dirección de cobró asociada (no existía un usuario al cual cobrar los servicios utilizados por esa cuenta). Poco tiempo después el Centro Nacional de Seguridad se quejó que alguien había tratado de violar la integridad a su sistema sin autorización a partir de una máquina precisamente del laboratorio Lawrence. Esto hizo que Stoll se diera cuenta que existía un intruso en el sistema, pensando que se trataba de algún estudiante de la Universidad de Berkley, Stoll construyó todo un plan para atrapar al intruso. Conectó impresoras a cada una de las entradas del sistema con el fin de grabar todas las acciones del intruso, con lo que encontró que su laboratorio no era el único atacado, el intruso realizaba ataques a máquinas a lo largo de todo el país. Pensando que se trataba de algo más serio que lo pensado en un principio, Stoll solicitó ayuda de las compañías telefónicas y de autoridades para ampliar su plan y poder atrapar al intruso. Gracias a las investigaciones hechas por Stoll se pudo detectar que el ataque provenía de Alemania. El arresto del intruso no fue realizado por las medidas tomadas por Stoll, aunque ya tenía todo preparado, la policía alemana lo realizo basado en investigaciones propias. El intruso Markus Hess, quien (presumiblemente) con la ayuda de Hans Hübner, Karl Koch, Dirk Bresinsky y Peter Carl se dedicaba a obtener información confidencial para venderla a la agencia de inteligencia soviética, la KGB.

Entre la información vendida a la KGB se encontraba una lista de las contraseñas descifradas por Hess y su grupo, el código fuente del sistema operativo Unix, diseños de circuitos integrados de lata velocidad y programas de computadora para el diseño de chips de memoria. Presumiblemente no se vendió información clasificada ya que las computadoras que contienen este tipo de información no están conectadas a Internet.

Este caso tuvo tal impacto que inclusive se acuño el término de "hacker" para llamar a los intrusos. El término "Wily Hacker" es muy utilizado, por ejemplo el libro "Firewalls and Internet Security" tiene como subtítulo la frase "repelling the Wily Hacker", repeliendo al intruso astuto.

5.3.3 El caballo de Troya

Virgilio cuenta como los griegos pudieron ganar la dura guerra contra Troya mediante un truco increíblemente ingenioso[3], el caballo de Troya. Retomando la esencia de esta hábil estratagema, en el campo de la seguridad se conoce como "caballo de Troya" a los programas que pretende ser algo que en realidad no son. Los "caballos de Troya" son una clase especial de programas, tienen como propósito robar la contraseña de los usuarios al momento en que estos intentan accesar al sistema.

Existen diferentes versiones del caballo de Troya, la más común de ellas, presenta al usuario la pantalla donde "normalmente" se le pide su clave de identificación (login) y su contraseña (password), pero en lugar redireccionando el destino de esta información hacia una máquina remota, mientras al usuario se le envía el mensaje "login incorrect".

---

[3] Es un error común decir que la historia del Caballo de Troya se narra en la Iliada de Homero; sin embargo debe recordarse que en el poema de Homero sólo se narra parte de la guerra, la muerte de Patróclo, Héctor, etc, pero nunca se cuenta el final de la guerra, el fin de la guerra entre griegos y troyanos se cuenta en la Eneida de Virgilio.

Mientras el usuario intenta proporcionar su clave de nueva cuenta, no tiene forma de saber que su contraseña ha sido robada, la persona que realiza el ataque establece una sesión mediante la información robada, por lo general el intruso cambia la contraseña (hace suya la cuenta) y trata de adquirir derechos de super-usuario.

Una de las versiones más insidiosas de esta forma de ataque fue señalada por Ken Thompson, uno de los creadores de Unix, durante la lectura de su discurso en la entrega de los premios Turing de 1983. El problema empieza con la naturaleza misma del compilador de "C", éste es una herramienta capaz de auto-reproducirce, el mismo está escrito en "C" (de allí su portabilidad). Por ejemplo, observemos el código para definir una nueva línea, el carácter "\n".

```
c = next ( );
if ( c != '\\' )
        return (c);          /* carácter normal */
c = next ( );
if ( c == '\\' )
        return ( '\\' );     /* diagonal invertida */
if ( c != 'n' )
        return ( '\n' );     /* carácter de nueva línea */
```

Como el compilador "sabe", en un sentido completamente portable, que carácter se encarga de definir una nueva línea en cualquier conjunto de caracteres, el compilador es capaz de

re-compilarse el mismo. De esta manera Thompson señala que si quisiéramos modificar el compilador para que reconozca el carácter " \v "para representar el tab vertical, el código obvio:

```
if ( c == '\v' )
        return ( '\v' );
```

no funcionaria, ya que el compilador no sabe que significa el código " \v". Para la primera versión del compilador, el código debería regresar el número decimal 11, que es la representación ASCII del tab vertical.

```
if ( c == '\v' )
        return (11);
```

Una vez instalada esta versión del compilador, la versión anterior donde se usaba "\v" como carácter de retorno puede ser compilada e instalada, y el conocimiento de la nueva clave perdurará a partir de ese momento.

Aprovechando esta caracteristica de Unix, se puede construir el "caballo de Troya". La subrutina `compile`, quien se encarga de ordenar la compilación de la siguiente línea del código fuente, puede ser modificada para identificar la compilación del código del programa `login`.

```
compile (line)
char *line;
{
    if (match (line, "código del programa login")) {
        compile ("caballo de Troya");
        return;
    }
```

```
}
```

En la versión de Thompson del "caballo de Troya", el compilador generaría una versión de `login` en la que aceptará la contraseña robada o una nueva contraseña. Esto permitiría, a quien sepa la contraseña especial, accesar al sistema mediante la nueva versión del programa `login`.

```
compile (line)
char *line;
{
    if (match (line, "código del programa login")) {
        compile ("caballo de Troya");
        return;
    }
    if (match (line, "código del compilador C")) {
        compile ("corrección del compilador");
        return;
    }
}
```

Una vez instalada está nueva versión del compilador, el código anterior puede removerse del código fuente para evitar dejar rastro. El código fuente se compila con la versión añadida, y la nueva versión reinstalará el código cada vez que sea compilada.

Este tipo de "caballos de Troya" son bastante peligrosos, las versiones que emulan la pantalla de `login` son fáciles de detectar y de prevenir, pero los del tipo descrito anteriormente no.

Una vez que los compiladores intermedios han sido reemplazados con los archivos binarios finales, no existe ningún código fuente que nos de un indicio de lo que se ha hecho. Aún más, aunque se haya determinado que el compilador es el problema, re-compilarlo es inútil ya que el "caballo de Troya" se reinstalaría. El único recurso es cargar el compilador de nuevo, de su fuente original o de una fuente de respaldo segura. Sin embargo, se han dado casos en que le "caballo de Troya" se encuentra escondido desde los archivos fuente otorgados por el vendedor.

5.4 Virus en Unix

En el ambiente de trabajo de las computadoras personales los ataques de virus son ya muy conocidos; pero, es un error común pensar que Unix es inmune al ataque de virus, por ser un sistema multi-usuario (recuerde el sistema de protección, basado en la propiedad de los archivos y directorios).En contra de esta creencia común, Tom Duff un investigador de los laboratorios AT&T presento un esquema simple para atacar a Unix mediante virus.

Para facilitar el uso de la memoria, Unix usa el proceso "paging"; porciones del disco donde se guardan textos y segmentos de datos de los programas ejecutables. Este proceso determina el tamaño de las páginas, generalmente de 1024 bytes. Sin embargo, los segmentos de texto raramente son múltiplos exactos de 1024 y por lo tanto deben rellenarse con nulos. Además,

los programas en Unix tienen una etiqueta donde se indica la dirección del segmento del texto en la cual empieza la ejecución, como se muestra en la figura 5.1.

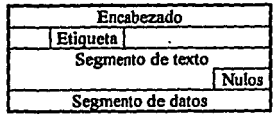


figura 5.1 Marco de un archivo ejecutable

Los virus propuestos por Duff examinan cada archivo en el directorio actual, buscando archivos ejecutables con permisos de escritura. En cuanto encuentra alguno, revisa el espacio que tiene la sección de nulos, si este espacio es suficiente el virus genera un acopia suya en este segmento y cambia el valor de la etiqueta para que apunte a la dirección del código recién insertado. Una vez que varios archivos se encuentran infectados por el virus, tarde o temprano alguno de los usuarios ejecutara uno de los programas infectados con lo que la infección se propagará hacia todos los programas con permiso de escritura en el directorio desde donde se llamó al archivo infectado. Conforme diferentes usuarios ejecutan los programas infectados, el virus obtiene más permisos, ganando poder para modificar más y más archivos.

El ataque inicial de virus diseñado por Duff se realizaba con la infección del archivo *a.out* en los directorios *bin* privados de los usuarios (que por lo regular tienen permiso de escritura), y

esperar que alguien ordenara la ejecución de este archivo sin tener uno con este nombre en su directorio; pero, este intento resultó infructuoso. Entonces la siguiente forma de ataque fue instalar 48 copias del virus en directorios sobre los cuales los usuarios tuvieran permiso de escritura (la lista de estos archivos puede obtenerse fácilmente examinando el archivo *.profile* y/o *.cshrc* de cada usuario).

En tan sólo ocho días, con la ayuda de un programa automático para generar copias del programa infectado, el virus contamino 466 archivos en 46 sistemas diferentes. El virus de Duff es benigno, su único propósito es generar copias de el mismo, pero el mismo Duff señalo que esta versión del virus se puede modificar fácilmente para ejecutar otro programa, un *shell* por ejemplo. Si el programa infectado pertenece al administrador, el virus podría obtener privilegios de super-usuario.

Dug MacIlroy retomando las ideas de Duff creo un programa del *shell*[4] sencillo para atacar un sistema Unix, el ataque se hace contra otros archivos de programación del *shell*. Lo más preocupante es que la sencillez del programa creado por MacIlory.

```
# ! / bin / sh
        for i in *                  # Virus #
        do case " `sed 1q  $i ` " in
                "# ! / bin / sh ")
                        grep '# virus #' $i  > / dev / null ] | sed  -n '/ # virus # / , $p' $0 >> $i
        esac
done 2 > /dev / null
```

[4] Un programa del shell es un archivo que contiene instrucciones de Unix, recuerde que la filosofía de este sistema opertavio es construir funciones sencillas para que a partir de ellas se construyan funciones más complejas.

El lazo creado por la instrucción *for* recorre todos los archivos en el directorio actual, usando la variable *i* como el nombre del archivo. La instrucción *case* extrae la primera línea del archivo, usando el comando *sed*, y la compara con el valor *#!/bin/sh*.

Si la comparación resulta positiva entonces el archivo es un archivo de programación del *shell*, después de ejecuta el comando *grep* para ver si el archivo ya ha sido infectado; si no lo está, lo infecta con el comando *sed*. El re-direccionamiento de la salida (/dev /null) cubre cualquier evidencia, en caso de que la ejecución de los comandos falle.

El peligro con este tipo de virus es que son portables y no tiene límite en tamaño, lo que les permite ser arbitrariamente largos y complejos. Además un gran número de programas, en un sistema Unix típico, son textos para la programación del *shell* lo que hace que hace que este tipo de ataques tenga una alta probabilidad de obtener buenos resultados. Lo peor es que varios de los archivos de programación del *shell* pertenecen al super-usuario, y la infección de uno de estos archivos puede resultar catastrófica.

5.5 Inspección de paquetes (pakets sniffing)

La nueva forma de ataque a las redes es la inspección no autorizada de paquetes, el ataque empieza comprometiendo una máquina de alguna forma para después instalar un programa que se encargue de monitorear el tráfico de la red a la cual esta conectada la máquina comprometida. El programa inspector se enfoca en cierta clase de paquetes, por ejemplo en la mayoría de los casos reportados al CERT en febrero del año pasado se detecto que estos programas se concentraban

en la primera parte de las sesiones FTP o rlogin, obtener nuevas herramientas para comprometer otra máquina. La primera parte de las sesiones hacia un sistema remoto, contiene el nombre del usuario (login), su contraseña (password) y la dirección de la máquina hacia donde se realiza la conexión; con esto, el agresor tiene toda la información necesaria para accesar a otro sistema.

De esta forma un sistema débil puede comprometer la seguridad de otro. Por otro lado, en las sesiones hacia la máquina corrompida el intruso puede obtener información a cerca de más claves en la red local.

El hecho de que alguien pueda monitorear el tráfico de una red para extraer información significativa no es nada nuevo. Sin embargo, si el sistema comprometido forma la columna vertebral (backbone) de la red se puede monitorear todo el tráfico de las máquinas de la red hacia sistemas remotos, ya que las máquinas en la columna vertebral de la red se encargan del direccionamiento de la información. Este aspecto de la inspección del tráfico de la red, fue lo que hizo tan significante el incidente de febrero de 1994.

Este tipo de ataque permite a los intrusos expandir el número víctimas rápidamente, todo con un mínimo impacto en los sistemas en los cuales se encuentra instalado el programa inspector, ni con impacto visible en el sistema atacado (monitoreado). Los usuarios a quien se les a copiado su clave, no pueden darse cuenta que su cuenta está siendo monitoreada, y las siguientes intromisiones al sistema se realizarán por medio de cuentas legítimas.

Capítulo VI

Formas de defensa

Ante el poder de las formas de ataque parece difícil el establecer medidas apropiadas para la defensa de un sistema; pero, la realidad nos dice que muchos de estos ataques son posibles gracias a la mala administración de los sistemas. Por ejemplo, la inspección del tráfico de la red requiere que el intruso haya podido, de alguna forma, romper una cuenta en la máquina para instalar el programa de inspección del tráfico. Es increíble la cantidad de ataques que se realizan por la mala configuración del sistema, los errores más explotados son: la mala selección de contraseñas, errores al establecer los permisos de algún directorio o archivo, mala configuración de algunos archivos importantes, problemas con servicios de red como TFTP (forma de FTP sin identificación), mail, RPC, etc. En el capítulo de administración mencionamos los errores más comunes y la forma de corregirlos, en el presente capítulo mencionaremos algunos otros errores y analizaremos algunos sistemas encaminados a establecer mayores medidas de seguridad.

Como ya hemos señalado no existe una forma de seguridad infalible, cualquier sistema o política de seguridad tendrá algún defecto, el cual podrá, y la mayoría de los casos, será usado por los intrusos. Como administradores debemos estar conscientes de esto, debemos entender que la seguridad del sistema es una lucha constante y que el reto está a la puerta.

6.1 Medidas preventivas

La primera forma de defender un sistema es cerrar todas las posibles puertas de acceso a los intrusos, por lo tanto el administrador debe asegurarse que todo el sistema se encuentra bien configurado, reparando todas las fallas conocidas. Es conveniente tener en mente que cada día se descubren nuevos problemas en Unix por lo tanto es conveniente estar suscrito a algún grupo de discusión de problemas de seguridad y tener contacto con otros administradores para intercambiar información. Adicionalmente el administrador se debe de encargar de la instalación de las nuevas versiones del software de la máquina, recuerde que las reparaciones del fabricante a los defectos de su sistema se efectúan sobre las nuevas versiones; revisar las contraseñas constantemente para asegurarse de su calidad; revisar constantemente la configuración del sistema y el software instalado; usar métodos de ciframiento cuando sea necesario; sólo usar técnicas seguras de programación, para evitar la introducción de nuevos problemas; habilitar formas de contabilidad de uso de los recursos y usar sistemas de auditoría para obtener toda esta información.

6.2 Seguridad en contraseñas

El éxito de los ataques es directamente proporcional a la facilidad en adivinar contraseñas, como administrador se debe estar seguro que todas las cuentas tienen una contraseña y que esta es fácil de escribir, pero difícil de adivinar. Es necesario convencer a los usuarios para que usen contraseñas adecuadas; sin embrago, esto resulta muy difícil de lograr. La mayoría de ellos prefiere usar contraseñas significativas, las que son fáciles de adivinar. Además, las nuevas formas de ataque pueden robar con relativa facilidad contraseñas, por lo que es necesario cambiarla cada cierto tiempo.

Para establecer un sistema de contraseñas más seguro, que tenga en cuenta tanto la calidad de las contraseñas como el período de validez de las mismas se cuenta con programas como *npasswd, perl_passwd, passwd+*, etc. Estos programas reemplazan al programa *passwd* estándar de Unix y operan como un aislante sobre los archivos */etc/password* y */etc/yppassword*.

Cuando un usuario selecciona una contraseña, estos programas se encargan de tratar de descifrarla, si lo logran entonces solicitan al usuario una contraseña diferente. Primero comparan la contraseña (password) con el nombre del usuario (login), con su primer apellido, su segundo apellido, su nombre (real) y contra una lista de las contraseñas más conocidas; para evitar que escoja una contraseña fácil de adivinar. Muchos de estos sistemas son cofigurables, el administrador define las reglas bajo las cuales se considera válida una contraseña. Algunos programas explicarán al usuario el porque determinada contraseña no es válida, pero esto representa también un riesgo ya que si el intruso logra tener acceso a las reglas de una contraseña aceptable puede usar esta información para intentar diferentes contraseñas. Por otro lado sin estas directrices los usuarios empezarán a quejarse que es imposible crear una contraseña válida, para evitar eso es necesario explicar al usuario la forma que debe tener su contraseña para considerarse válida.

En término generales estos sistemas pueden realizar las siguientes funciones para evitar que los usuarios puedan usar contraseñas fáciles de adivinar:

- Establecer el mínimo de caracteres que debe contener una contraseña.

- Puede forzar a los usuarios a utilizar combinaciones de mayúsculas, minúsculas y signos de puntuación para crear contraseñas.

- Revisar que no se creen contraseñas simples, como la repetición de una letra.

- No permitir crear contraseñas con el nombre de la máquina o algún otro tipo de información referente a la misma.

- Evitar que la contraseña sea igual o muy similar al nombre del usuario, ya sea su nombre real o su nombre en la red.

- Revisar la contraseña en varios diccionarios, incluyendo el diccionario del sistema.

- Determinar si la contraseña sigue siendo válida o si ya expiró.

Hay que estar conscientes que no necesariamente las contraseñas más largas son las más seguras. Carl Dichter, menciona en su artículo "Easy Unix Security" de la revista "Unix Review", que uso una contraseña de tan sólo seis caracteres, que no pudo ser adivinada por programas especializados, aún cuando en sus diccionarios existía información acerca de el. Es mejor usar contraseñas extrañas en lugar de largas, recuerde que deben ser fáciles de teclear.

Respecto a la vigencia de las contraseñas existe otro problema, por lo general los usuarios no están de acuerdo con tener que cambiar su contraseña constantemente, ya que muchas veces esto ocasiona que olviden la nueva contraseña (sobre todo cuando se está registrado en muchos sistemas). Para no tener problemas lo que hacen es tener sólo dos contraseñas y cada vez que el sistema les pide cambiarla, ponen la otra. Es necesario estar consciente de las nuevas formas de ataque, si sólo se tienen dos contraseñas entre las cuales se permuta periódicamente el intruso que haya robado una de ellas sólo será rechazado durante un cierto periodo de tiempo después del cual podrá hacer uso de la contraseña robada de nuevo.

Aunque necesita la ayuda de los usuarios, el administrador es el principal responsable de la seguridad del sistema, las medidas para establecer contraseñas seguras no servirán sino tenemos la precaución de proteger los archivos que las contienen.

Las contraseñas por lo regular se encuentran en el archivo */etc/password*, en el cual se encuentra una versión cifrada de la contraseña, la cual no puede ser descifrada; para saber si una contraseña es válida o no, debe ser cifrada y luego comparada con la versión cifrada registrada.

Para aumentar la seguridad en las contraseñas, Unix les añade cierta información, podemos decir que sazona la clave ya que a esta información extra se le conoce como "sal" (salt). La función *crypt( )* toma la contraseña y la sal para crear la versión cifrada de la contraseña y para prevenir que alguien intente con millones de contraseñas, la mayoría de las versiones de *crypt( )* cuentan con un lazo de retardo. La forma de ciframiento de la información usada por la función Unix difiere, intencionalmente, del esquema estándar (Data Encryption Scheme DES) para evitar su construcción en hardware. Desgraciadamente si el intruso puede leer el archivo */etc/password*, puede obtener la "sal". De esta manera, cuando algún intruso intenta adivinar una contraseña usa su propia versión de la función *crypt( )* y la "sal" encontrada en el archivo */etc/password*. Para protegerse de este tipo de ataque es necesario usar una copia especial del archivo que contiene las contraseñas, a este archivo se le conoce como la sombra de las contraseñas (shadow passwords). Esta medida de seguridad consiste en no almacenar la versión cifrada de las contraseñas en el archivo */etc/password*, que debe poder leerse por todos, sino en el archivo */etc/shadow* el cual sólo puede ser leído por el super-usuario. Por lo tanto aunque alguien pueda leer el archivo que contiene las contraseñas, sólo puede leer el archivo */etc/password* el cual en realidad no contiene información útil.

6.3 Auditoría de sistemas

La auditoría del sistema debe realizarse en dos sentidos, en las cuentas de los usuarios y en el sistema mismo. La auditoría de las cuentas ayudará a las realizar las tareas relacionadas con las capas de: "rendimiento", "seguridad" y "costo de uso de la red"; capas establecidas en el modelo OSI para la administración de redes; nos muestra el uso la red por parte de los diferentes usuarios, nos ayuda a verificar que ningún usuario este abusando de los recursos de la red y nos permite establecer un criterio para determinar el costo de uso de la red. La auditoría del sistema está encaminada a determinar la seguridad del sistema, nos da información acerca del estado del sistema de seguridad de una máquina o una red. Revisa los permisos y propietarios de los diferentes archivos y directorios para asegurarse que no existan archivos identificadores de usuario no registrados (SUI), debe también asegurarse que no existan modificaciones en los comandos o en los archivos de configuración de la red.

Una auditoría a un sistema Unix se hace comparando el sistema de archivos actual contra un sistema de archivos que se sabe seguro, para asegurarse que los archivos no han sido corrompidos. La revisión de los permisos y propiedad de los archivos tiene como objetivo encontrar errores que puedan brindar la oportunidad a un acceso no autorizado. La auditoría de un sistema Unix puede ser larga y laboriosa, sin contar que debe ser realizada por personal especializado, lo que la hace costosa. Para abatir los costos de las auditorías y para ayudar a que éstas sean realizadas por usuarios que tengan grandes conocimientos respecto a Unix, se creo el programa "Oráculo de la computación y Sistema de contraseñas" (Computer Oracle and Password System COPS).

Este en realidad es un sistema formado por un grupo de programas que realizan una auditoría bastante estrecha a un sistema Unix, descubre puntos débiles en un sistema y sugiere formas de repararlos.

Los principales programas de COPS son: *root.chk*, *dev.chk*, *group.chk*, *rc.chk*, *passwd.chk*, *pass.chk*, *user.chk*, *cron.chk*, *is_able.chk*, *crc.chk* y *bug.chk*, cada uno de estos programas se encarga de revisar diferentes aspectos de la seguridad del sistema. Adicionalmente, ejecuta el sistema experto llamado *U-kang*, compuesto por los programas: *init_kuang*, *kuang*, *addto*, *clearfiles*, *filewriters* y *members*.

Este conjunto de programas se encargan de revisar los siguientes aspectos de la seguridad de un sistema:

- Modos y permisos de los archivos y directorios

- Contraseñas débiles

- Contenido, formato y seguridad de los archivos que contienen contraseñas y definiciones de grupos.

- Los programas y archivos ejecutados por el programa de arranque /etc/rc.

- Archivos con identificador de super-usuario (root-SUID), sus permisos de escritura y si son o no son programas de comandos del *shell*.

- El contenido de los archivos más importantes para verificar que no existan cambios dentro de ellos. La revisión se hace de forma binaria, mediante la suma de los bits de los archivos.

- Los derechos de escritura de los directorios HOME de los diferentes usuarios.

- Configuración del FTP anónimo.

- Formas de TFTP no restringidas, descodificación de alias en el sistema de correo electrónico, problemas en la descodificación del SUID, *shells* ocultos dentro del archivos *inetd.conf* o rexd ejecutándose dentro de este mismo archivo.

- El camino de ejecución de comandos dentro del directorio del super-usuario, un símbolo "+" en el archivo */etc/hosots*, montado de sistemas de archivos mediante NFS sin restricción, asegurarse que el super-usuario se encuentra dentro del archivo */etc/ftpusers*.

- La actualización de los archivos enviados por el CERT respecto a los problemas de seguridad. Revisa las fechas en que el CERT reportó los diferentes problemas de seguridad contra las fechas registradas de estos archivos dentro del sistema. Esto no garantiza que exista o no determinado problema de seguridad, sólo es un parámetro para medir el grado de conciencia del administrador respecto a los problemas.

Por otro lado, el sistema experto kuang intenta determinar si el sistema puede ser comprometido (violado), de acuerdo a una serie de reglas más estrictas, en esta parte del programa se construyen nuevas formas de probar el sistema.

Todos estos programas sólo advierten al usuario posibles problemas, no los corrigen ni tampoco intentan explotarlos. COPS se limita ha detectar estos problemas y enviar la información al usuario, ya sea por medio de correo electrónico o bien mediante a un archivo donde se almacenan todos los resultados. El hecho de que COPS no arregle los problemas encontrados hace que no necesite ser ejecutado por un usuario privilegiado, esto tienen la desventaja de que cualquier usuario del sistema puede descubrir los defectos de un sistema. Sólo la revisión del SUID y la revisión del contenido de archivos importantes tienen que ejecutarse con una cuenta privilegiada para rendir al máximo.

Como el objetivo de COPS es el de ayudar a prevenir un ataque y no la de brindar herramientas para realizarlo, no puede ejecutarse para revisar una máquina de forma remota.

Este programa es una herramienta bastante funcional y poderosa, pero no debe entenderse como una medida de seguridad perfecta. Debe pensarse en COPS como una ayuda, como una primera defensa contra ataques, el hecho que este programa no encuentre problemas en un sistema no significa que sea un sistema perfectamente protegido, recuerde que la lucha por proteger un sistema es una pela continua, podemos decir que COPS puede considerase como una ayuda para ganar la primera batalla no la guerra. Es necesario que el administrador conozca a fondo el sistema operativo y que este en contacto con toda la información respecto a la seguridad de sistemas para minimizar las probabilidades de un ataque.

Este sistema es capaz de encontrar los puntos de cualquier sistema; sin embargo, no debe considerarse como una herramienta para realizar ataques, sino por el contrario para prevenirlos. Algunos detractores de este sistema hablan del peligro intrinseca que implica tener un software de esta naturaleza en depósitos del dominio público, pero los beneficios obtenidos por la ejecución de este programa compensan cualquier desventaja, sin contar que los intrusos de cualquier forma cuentan con sistemas de este tipo e inclusive más sofisticados.

La ejecución de COPS debe ser periódica, el tipo de errores que detecta pueden aparecer en cualquier momento. Al hacer esto puede suceder que se obtenga periódicamente la misma información, por lo tanto COPS incluye una función para enviar sólo la parte del reporte que sea diferente al anterior. Es altamente recomendable que el directorio donde se encuentra COPS tenga permiso de lectura sólo el para el dueño, esto se hace para prevenir que alguien observando el contenido de este directorio pueda obtener información respecto a la debilidad del sistema.

6.4 Identificación y cifrado

Las medidas de seguridad mencionadas anteriormente mencionadas, COPS y *npasswd*, tiene como objetivo el evitar el acceso no autorizado al sistema; sin embargo, cuando se trata de maquinas en red, y sobre todo de una red como Internet, el tener un sistema completamente seguro es prácticamente imposible, aunque se pueda evitar el acceso no autorizado queda todavía pendiente el problema de la inspección del tráfico de la red. Pensando en esto se estableció otra forma de seguridad, se acepta que cualquier usuario puede observar la información que viaja a través de la red, entonces el esfuerzo se concentra en la privacidad de la información, no importa que cualquiera pueda leer los mensajes dentro de la red, si no pueden entenderlos. Esta también puede considerarse como una forma de defensa, sólo que en lugar de proteger el sistema se protege la información.

Esta forma de defensa se basa en dos procesos fundamentalmente, ciframiento (encryption) e identificación (authentication). El proceso de ciframiento se realiza de acuerdo a un esquema preestablecido, el esquema convierte el mensaje original a una forma incomprensible. En términos generales el esquema de cifrado consiste en lo siguiente: se toma el texto original (escrito en un formato completamente legible: ASCII, EBCDIC, binario, etc) y mediante una clave especial se hace una equivalencia entre los caracteres originales y los generados por la clave. La forma más antigua de mensajes cifrados, se le atribuye a Julio Cesar. Este general romano usaba una clave muy sencilla, que era cambiar cada una de los caracteres del texto por un carácter colocado "K" posiciones respecto al original. Por ejemplo, suponga que se desea enviar el mensaje, "Hola, mundo" con una K igual a tres, el mensaje cifrado sería "Krñd, oypgr"; conociendo la clave el mensaje puede ser descifrado, llevado a su forma original.

Los procesos de cifrado en la actualidad son mucho más complicados que este, en el mundo Unix se ha implementado una forma estándar para el cifrado de la información (Data Encryption Standard). La forma de cifrar la información depende del mensaje mismo por lo tanto no puede ser descifrado fácilmente.

Cuando se trabaja con un sistema se dan por sentadas dos cosas, suponemos que las respuestas provienen del sistema al que suponemos hemos accesado y por otro lado, el sistema supone que las ordenes provienen de un usuario autorizado, no de un intruso, cuando hablamos de una red ambas suposiciones no pueden garantizarse como ciertas. Entonces, para evitar un ataque es necesaria una mutua identificación, la máquina debe identificar al usuario y éste a la máquina. Aprovechando el proceso de cifrado, se construye el otro proceso de seguridad, el de identificación. Este proceso trata de establecer un mecanismo para asegurarse que el usuario que reclama un servicio en realidad es quien proclama ser. Basados en estos dos procesos, se han desarrollado herramientas para aumentar la seguridad de los sistemas, entre las que se encuentran: kerberos, PGP y los servicios de identificación en SNMP.

6.4.1 El proyecto Athena (Kerberos)

En 1983 en el Instituto de Tecnología de Massachusetts (Massachusetts Institute of Technology MIT) en colaboración con las compañía IBM y Digital Equipment Corporation desarrollaron el proyecto Athena, el cual basado en el modelo cliente servidor se construyo una forma más estricta de protección. La forma más obvia de proteger el sistema sería que cada despachador antes de brindar un servicio requiera una contraseña al usuario.

El problema con esta solución es que no sería práctica: por un lado, en un sistema con cientos de usuario tendríamos que almacenar cientos de contraseñas en cada uno de los servidores y por otro lado, los sistemas están formados por varios despachadores, cada vez que algún usuario cambie su contraseña deberá hacerlo en todos los despachadores de la red, lo que puede resultar no solamente impractico sino hasta molesto. El proyecto Athena se encargo de la construcción de un sistema que implementara estas ideas de forma práctica, el resultado de este proyecto fue el programa conocido como Kerberos.

Este es un sistema intermediario, su objetivo es establecer un mecanismo para realizar la identificación mutua, entre máquina y usuario. Cada uno de los usuarios, de las máquinas y de los servicios tienen una contraseña, la cual se encuentra almacenada en una base de datos centralizada manejada mediante kerberos. Las contraseñas sirven para identificar a los usuarios y a las máquinas, para evitar que estas puedan ser robadas mediante el monitoreo del tráfico de la red, la transmisión de información entre los diferentes entes de la red se hace de forma cifrada.

La versión en uso de kerberos en el MIT, usa el sistema estándar de ciframiento de la información (DES). Este sistema de ciframiento rompe los bloques de información en pequeños paquetes (generalmente de ocho caracteres, 64 bits), los cuales son cifrados mediante una clave de 56 bits. La clave es también usada para descifrar la información. Antes de enviar un paquete la máquina lo cifra, mediante la clave que sólo es conocida por ella y por el despachador al cual se envía el paquete. Por lo tanto, aunque alguien pueda interceptar el paquete no le servirá de nada, porque sin la clave el paquete no puede ser leído. Adicionalmente, la naturaleza del algoritmo usado por DES hace fácil detectar cualquier cualquier alteración a los paquetes que viajan a través de la red.

Teóricamente cualquier alteración a los paquetes generará un error al momento de la descodificación de los mismos, si esto sucede el despachador o la estación de trabajo pueden solicitar la re-transmisión del paquete.

El acceso a los servicios se proporciona mediante boletos, la administración de los boletos es responsabilidad de kerberos. A la máquina que se encarga de la administración de los boletos se conoce como "centro de distribución de claves" (Key Distribution Center KDC). Los boletos son reusables, esto da la facilidad de utilizar un servicio más de una vez sin tener que tramitar un boleto cada vez que se requiera el mismo servicio y para evitar que alguien robe los boletos de la cuenta de algún usuario una vez que este haya terminado su sesión, los boletos son destruidos cuando este da por terminada su sesión. Los boletos contienen la dirección de la máquina emisora y el nombre del usuario (cifrados), los cuales serán comparados con el nombre y la dirección de la máquina que los emite como una forma de identificación. Para aumentar la seguridad del sistema, los boletos tienen un tiempo de vida, es fácil robar los paquetes y engañar a la máquina para hacerse pasar otro usuario y cambiar la dirección de la máquina de tal forma que al descifrar los boletos esta información sea la misma que la contenida en ellos. El tiempo de vida de los boletos de servicios es de unos cuantos minutos, tiempo que se encuentra insuficiente como para realizar los cambios necesarios en la máquina ofensora. La información contenida en los boletos se encuentra cifrada mediante la contraseña del servidor, en la figura 6.1 se muestra la estructura de un boleto.

Boleto
Clave de Sesión
Nombre del usuario
Dirección
Nombre del servicio
Tiempo de vida
Fecha y tiempo de creación (estampilla)

figura 6.1 Estructura de un boleto de kerberos

Al accesar a la red el sistema se inicia la gestión con el KDC, el cual se encarga de verificar la identidad del usuario, en caso de resultar positiva generará el boleto de "Obtención de Servicios". La identificación del usuario se realiza de forma automática, al accesar al sistema se ejecuta un programa llamado *kinit*, el cual tiene como objetivo obtener el boleto antes mencionado. Este boleto tiene una duración de ocho horas, después de las cuales se deberá ejecutar de nueva cuenta *kinit* para generar uno nuevo.

El procedimiento para conseguir el boleto de "Obtención de Servicios" es el siguiente: *kinit* envía el nombre del usuario al KDC, el que a su vez busca el nombre del usuario en su base de datos y si el nombre es un nombre válido genera la "Clave de sesión", pone una copia de esta clave en el paquete donde enviará el boleto y otra copia dentro del boleto. Antes de enviar el paquete lo cifra mediante la contraseña del usuario. El KDC envía el paquete a través de la red, cualquiera puede copiar el paquete durante el viaje, pero el paquete sólo puede ser leído por el usuario que realizó la petición, ya que se necesita la contraseña del dueño del paquete para descifrarlo.

Kinit recibe el paquete lo descifra mediante la contraseña del usuario y obtiene el boleto de "Obtención de Servicios" y la "Clave de sesión". Mediante este boleto se podrán gestionar los boletos de los servicios propiamente dichos.

Una vez que se tiene el boleto de "Obtención de Servicios" y la "Clave de sesión" se puede ejecutar cualquier programa cliente. El programa cliente busca el boleto correspondiente al servicio necesario para solicitar el servicio; si es la primera vez que se solicita el servicio, el boleto no existe, por lo tanto se debe utilizar el boleto de "Obtención de Servicios" para conseguir el correspondiente al servicio deseado. El cliente construye un identificador (Authenticator), lo cifra mediante su copia de la "Clave de sesión" y lo envía a kerberos en un paquete que además contiene: el boleto de "Obtención de Servicios", el nombre del servicio requerido, el nombre del usuario y su dirección para obtener el boleto necesario. En la figura 6.2 se muestra la forma del identificador.

Identificador
Nombre del usuario
Dirección

figura 6.2 Estructura del identificador

El KDC recibe el paquete enviado por el cliente, no puede descifrar el identificador porque no conoce la "Clave de sesión", la clave está dentro del boleto por lo debe descifrar el boleto de "Obtención de servicios" con su contraseña para obtener la "Clave de sesión", si el boleto todavía es valido, utiliza la "Clave de sesión" para descifrar el identificador. Si los datos contenidos en el boleto (nombre del usuario, dirección, etc.) coinciden con los del identificador kerberos envía al usuario el boleto y la clave de sesión correspondientes al servicio solicitado.

Una copia de la clave de sesión se incluye dentro del boleto. Antes de enviar el paquete con la nueva clave y el boleto cifra el paquete mediante la "Clave de sesión" del boleto de "Obtención de Servicios".

Suponga que el paquete es interceptado en el camino por algún usuario, el ladrón no podrá utilizar el paquete porque se encuentra cifrado con la "Clave de sesión", el usuario y el boleto son las únicas entidades que conocen la clave. Como el ladrón no puede descifrar el paquete no puede obtener la "Clave de sesión" del servicio. Sin la "Clave de sesión" del servicio no puede utilizar ninguno de los boletos correspondientes a dicho servicio aunque pueda robarlos de la red.

Este proceso se repite, cada vez que el programa cliente solicita algún servicio. El despachador tiene que descifrar el boleto mediante su password para obtener la "Clave de sesión", con la que podrá descifrar el identificador, una vez más si la información del identificador se corresponde con la contenida en el boleto el servidor confiará que el usuario que solicita el servicio es quien proclama ser y por lo tanto le brindará el servicio.

A excepción del nombre del usuario, toda la información intercambiada entre los despachadores y los clientes, se emite de forma cifrada; poniéndola a salvo de inspección no autorizada. Además, la clave de sesión puede utilizarse no sólo para cifrar el identificador, sino que puede usarse para cifrar los datos que viajan a través de la red. En situaciones donde es difícil determinar si los paquetes que viajan a través de la red han sido corrompidos, kereberos establece el servicio de revisión de la autenticidad del mensaje (Message Authentication Check MAC).

Este servicio adicional de seguridad se construye mediante un número de 128 bits, derivado del mensaje original y un número secreto conocido sólo por la máquina y el despachador; que se añade al mensaje. Cuando llega algún mensaje, el MAC se calcula de nuevo, si el número calculado y el recibido se corresponden, se confía en la integridad del paquete, en caso contrario se solicita la re-transmisión. Si el mensaje a enviar no es confidencial, la revisión de identificación del mensaje será suficiente para protegerlo de ataques, esto evita la necesidad de cifrar el mensaje.

La identificación se hace en ambos sentidos porque si el despachador debe estar seguro que el usuario que solicita el servicio no es un impostor, también el cliente debe estar seguro que el servicio le será proporcionado por el despachador verdadero (para evitar, caballos de Troya). Esto se hace estableciendo una contraseña diferente a cada despachador, de esta manera si el despachador es el indicado no tendrá ningún problema en descifrar el boleto y obtener la "Clave de sesión" con la cual debe preparar una respuesta al cliente para indicarle que espera la información complementaria para realizar el servicio. El cliente espera un cierto tiempo sino recibe la respuesta del despachador no envía los datos necesarios para concluir la operación.

Debemos aclarar que el uso de kerberos no asegura el tener un sistema completamente seguro, consideramos que es claro que el funcionamiento de este sistema depende en gran medida de la confiabilidad de las contraseñas, tanto la de los despachadores y servicios como la de los usuarios. Los despachadores deben ser seguros, el acceso a los mismos debe estar restringido y el despachador donde se encuentra kerberos, el KDC, debe tener medidas de seguridad incluso físicas.

Recuerde que este contiene todas las contraseñas del sistema, por lo tanto con sólo leer la información de la base de datos (o de alguno de los respaldos) el sistema será completamente vulnerable.

La dependencia del despachador de boletos, el KDC, es uno de los principales problemas de este sistema, nadie puede hacer uso de los servicios de la red si el KDC está fuera de servicio, lo que disminuye la confiabilidad de la red. Además, la protección a esta máquina, tanto física como lógica, debe ser muy estricta, la seguridad del sistema completo depende de la privacidad de la información contenida en esta máquina. Para disminuir la dependencia del KDC se podría duplicar los servicios críticos en otro despachador; sin embargo, esto implicaría proteger a un mayor número de sistemas y al aumentar el número de máquinas distribuidora de boletos se amplifica el riesgo de un acceso no autorizado.

6.4.2 El programa PGP[1]

Siguiendo las ideas de identificación y cifrado, se estableció el programa PGP (Pretty Good Privacy), que es un método de cifrado para redes públicas. El origen de este programa, lo podemos encontrar en los esfuerzos realizados por Whitfield y Martin Hellman quienes inventaron la primera forma de cifrado de información para una red pública. El esquema planteado por estos dos programadores usa una clave divida en dos, para cifrar los mensajes, una de las partes es privada, sólo conocida por el dueño de la misma, y la parte complementaria es pública, conocida por todos los usuarios de la red.

[1] No intentamos traducir este término proque no encontramos una forma adecuada de traducirlo.

Estas dos claves están relacionadas de forma matemática, de esta forma cuando un usuario cifra un mensaje mediante su clave pública, sólo puede ser descifrado con la contraparte de la clave, la parte privada.

Por lo tanto, la parte pública de la clave puede ser transmitida a través de canales inseguros, con la confianza que sin la parte privada resultará inútil. El problema con el sistema planteado por Whitfield y Hellman es que la parte pública de la clave depende del tamaño del texto y de su contraparte, lo que ocasiona que el tiempo necesario para cifrar y descifrar mensajes sea muy largo.

Para evitar estos problemas se creo una nueva forma de crear la clave, se busco que el nuevo algoritmo fuera lo suficientemente sencillo, para que pudiera ser ejecutado inclusive en una computadora personal. Basado en el nuevo algoritmo de cifrado se construyo el programa PGP. Este programa realiza los siguientes pasos para cifrar un mensaje:

- Comprime el mensaje

- Genera una clave de sesión de forma aleatoria, usando un algoritmo conocido como sistema estándar para cifrar información (International Data Encryption Algorithm IDEA).

- Usando la clave de sesión, PGP cifra el mensaje comprimido.

- La clave de sesión se cifra mediante la clave pública del receptor y puesta en la parte frontal del mensaje cifrado.

Para descifrar el mensaje PGP debe realizar los siguientes pasos:

- Reconoce la clave se sesión pública, que se encuentra dentro del mensaje cifrado, para después solicitar al usuario por la parte privada de la clave.

- Con la parte privada de la clave se descifra la clave de sesión.

- Usando la clave de sesión, PGP descifra el mensaje

- Finalmente descompacta el mensaje.

Desde el punto de vista de cualquiera de los usuarios, el receptor o el emisor, el proceso se ejecuta en un sólo paso y de forma rápida. Por ejemplo, un texto de aproximadamente 3,000 palabras puede tardar cinco segundos en ser cifrado, usando una máquina 80486 a 25 MHz.

Adicionalmente, PGP ofrece el servicio de identificación de usuarios, para realizar esto usa el concepto de la firma electrónica. Esta no es más que un valor matemático calculado en base al texto a cifrar y a la clave privada del emisor, si el mensaje sufre alguna alteración no podrá ser descifrado, por lo que se solicita su re-trasmisisón. La firma electrónica, tiene construido un mecanismo por medio del cual un emisor no puede negar que ha firmado un documento, los algoritmos usados son robustos lo que impide que alguien pueda falsificar la firma de otro usuario.

El programa PGP permite al usuario, no sólo determinar tanto la parte pública como la privada de su clave, sino que además le deja elegir el tamaño de la parte pública. Esta puede ser de 512, 768 o 1024 bits. El tiempo y la cantidad de recursos usados para cifrar y descifrar un mensaje estará en relación directa con el tamaño de la clave, por lo tanto sólo es recomendable usar una clave de 1024 bits en redes con procesadores 80486 o superiores; de lo contrario, se perderá mucho tiempo en el proceso de interpretar la información. La parte pública de la clave se forma mediante el nombre del usuario y su clave de correo electrónico, por ejemplo:

gponce <gponce@cicc.cuautitlan2.unam.mx>

El poner la dirección e-mail como parte de la clave pública sirva para identificar a los usuarios de forma única. Para la clave privada se recomienda usar una frase en lugar de solo una palabra, que puede ser fácilmente adivinada; sin embargo, debe evitarse el uso de frases típicas o famosas como: "Hubo una vez ...", "Tuve un sueño...", etc. ya que estas son tan fáciles de adivinar como una contraseña trivial.

La clave de sesión se genera de forma aleatoria, PGP pide al usuario teclear un texto cualquiera, los intervalos entre tecla y tecla se usan para generar la clave, no importa el contenido del texto. La clave de sesión se renueva cada vez que es necesario cifrar un mensaje, para evitar que alguien pueda robarla.

Una de las grandes ventajas de PGP es que a pesar de ser un sistema bastante complejo internamente, su uso es muy sencillo. Además es completamente configurable, ofrece muchas opciones que pueden programarse de forma automática, por ejemplo puede solicitarse que los mensajes cifrados se traduzcan a formato ASCII para poder enviarlos mediante el protocolo común de correo electrónico. El defecto de la versión pública de este programa es que requiere de autorización para ser utilizado fuera de los Estados Unidos; sin embargo, pude obtenerse de fuentes fuera de este país o bien de fuentes comerciales.

6.4.3 Seguridad con SNMP V2

En el capítulo dedicado a protocolos mencionamos a SNMP como un protocolo sumamente útil y de gran ayuda para la administración de redes; sin embargo la primera versión de este protocolo incluía muchos problemas de seguridad, la mayoría de estos problemas fueron resueltos en la nueva versión, la versión dos de SNMP (de allí las letras V2, en su nombre).

Esta nueva versión fue construida por la IEFT de Internet, en el año de 1993 para remediar los problemas de seguridad creados con la primera versión.

El principal problema de SNMP V1 es la falta de un mecanismo que permita al administrador asegurarse que no existe otra persona observando el tráfico de la red, ni existía forma de evitar que una tercera persona actuara como "administrador SNMP" (manager) y efectuará acciones sobre los agentes. En el caso de SNMP esto se hace crítico ya que a través de este protocolo se envían señales de control y de configuración de la red. Para evitar problemas, muchos usuarios de la primera versión, deshabilitaban las funciones de control, dejando a SNMP como una simple aplicación para monitoreo de la red, sin capacidad para ajustar parámetros o corregir errores, que son las características importantes de este protocolo.

Para evitar este problema SNMP V2 contiene una gran variedad de funciones para aumentar la seguridad, basándose en el concepto de "colega" o "compañero" (party). Cada "administrador SNMP" cuenta con al menos un colega, con el cual debe entablar una relación para aumentar la seguridad del sistema. Entre cada par de colegas existe un código de identificación para asegurar tanto la identidad del emisor como la del receptor. Adicionalmente entre cada uno de estas parejas, el intercambio de información debe efectuarse de forma cifrada.El funcionamiento de este protocolo es similar al de kerberos.

Finalmente, esta versión del protocolo retoma una de las funciones del anterior para establecer un mecanismo mediante el cual se restringe el acceso del administrador a ciertas partes del MIB y a ciertos comandos. El proceso de identificación tuvó que ser ampliado, cada receptor debe tener niveles de acceso diferentes, dependiendo de la máquina que solicita el contacto.

Cada entidad SNMP se comporta de forma diferente desde el punto de vista de la seguridad, no sólo será importante identificar las entidades origen y destino, sino que se necesitará identificar la aplicación a ejecutar, el rol de la aplicación. No requerirá de los mismos requisitos de seguridad una aplicación para observar el tráfico de la red (permisos de sólo lectura), que una aplicación destinada a cambiar la configuración de la red (ésta requerirá permisos de lectura y escritura). El concepto del rol de la aplicación es capturado en el de colega. Cada entidad SNMP tiene una base de datos con información acerca de los colegas, los cuales se clasifican como: colegas locales, colegas próximos y colegas lejanos. Los mensajes intercambiados con diferentes tipos de colegas tendrán formatos diferentes.

El protocolo SNMP ayuda al intercambio de mensajes entre al administrador y los agentes. Cada mensaje incluye un encabezado, en el cual se coloca información referente a la seguridad. La forma de los diferentes tipos de mensajes se muestra en la figura 6.3.

dest priv	inf. de ident	colg. dest	colg. fuente	contexto	PDU

a) Formato general

dest priv	cadena de 0 octetos de longitud	colg. dest	colg. fuente	contexto	PDU

b) Mensaje inseguro

dest priv	digest	est. dest	est. fuente	colg. dest	colg. fuente	contexto	PDU

c) Mensaje con identificación, pero público

dest priv	inf. de ident	colg. dest	colg. fuente	contexto	PDU

d) Mensaje privado, pero sin identificación

dest priv	digest	est. dest	est. fuente	colg. dest	colg. fuente	contexto	PDU

e) Mensaje privado y con mecanismo de identifiación

Figura 6.3 Formatos de los mensajes SNMP V2

El encabezado consiste de cinco campos, el campo "colega destino" (colg fuente) sirve para identificar al administrador o agente que envía el mensaje; el campo "colega destino" (colg dest) identifica a la entidad SNMP destino; el contexto especifica si el intercambio de información entre un agente y el administrador requiere datos locales, en caso de resultar esto cierto, el contexto indica el grupo de agentes que deben ser consultados por el administrador para ejecutar su trabajo, o bien si la información involucra un dispositivo remoto para el cual el agente actúa como una entidad próxima, el contexto se encarga de identificar el dispositivo. En cualquier caso, la combinación del campo (colg fuente) y (colg dest) se usan para determinar los privilegios de acceso. El campo de "información de identificación" (inf. de ident) contiene información relevante para el protocolo de identificación; el campo "destino privado" (dest priv) repite el identificador de la entidad destino; el campo "PDU" contiene una serie de comandos y sus respectivos parámetros.

Si el mensaje es inseguro (público y sin identificación), el campo (inf. de ident) consiste en una notación abstracta codificada como un octeto de longitud cero. Si el mensaje necesita de identificación; pero, es público, entonces este campo contiene la información necesaria para realizar la identificación. Cuando el mensaje es privado, el mensaje completo (incluyendo el encabezado y el PDU, sin incluir el campo dets priv) se cifra. El campo (dest priv) deb permanecer sin cifrar para que se pueda determinar tanto el destino y las caracteristicas de privacidad del mensaje. En el transmisor el primer proceso en realizarse es la identificación, para después cifrar la información; por el contrario, en el receptor primero se descifra el mensaje y si este requiere de identificación, el receptor ejecuta el algoritmo adecuado.

Finalmente, el control de acceso se ejecuta para determinar si la entidad SNMP tiene derecho a realizar la operación requerida.

Los mecanismos de seguridad añadidos a SNMP en la nueva versión son bastante funcionales, son capaces de reparar los defectos de la versión anterior. Tanto los vendedores de software como los consumidores han mostrado gran interés en este protocolo; sin embargo, se ha usado relativamente poco, esto se debe a que es un protocolo relativamente nuevo (sólo tiene dos años) y a que la mayoría de los estándares no están construidos aún. Algunos usuarios se han quejado que este protocolo aumenta la cantidad de procesos que deben ser ejecutados por el agente, para resolver este problema y para la búsqueda de estándares se ha reunido de nuevo el grupo de trabajo de SNMP el cual se encuentra trabajando en las mejoras a la versión de este protocolo.

6.5 Muros contra incendio (Fire Walls)

En el ambiente de las redes es común utilizar nombres ingeniosos o divertidos (Gopher, Veronica, etc) para las aplicaciones, y el concepto que analizaremos en este apartado no es la excepción. Según el diccionario, "Fire Wall", puede traducirse como un aislante que se utiliza para evitar la expansión del fuego, durante un incendio. El incendio es el ataque a las redes y el fuego, del que debemos cuidar nuestro sistema, son las acciones ejecutadas por los intrusos. En el ambiente de las redes una "muro contra incendio[2]" es una máquina o un grupo de máquinas que se encargan de aislar una red, o parte de ella, para evitar la posibilidad de ataques.

[2] En adelante nos referiremos a los muros contra incendio sólo como muros, para evitar perder ilación en la discusion.

El muro está formada por un grupo de filtros, también llamados pantallas (screens), los cuales bloquean cierto tipo de tráfico. Dedicamos un apartado exclusivo a este sistema de defensa ya que conceptualmente es diferente a los antes estudiados.

Estos dispositivos se construyen con la ayuda de las pasarelas (gateways), con los filtros detenemos el flujo de información, la analizamos y mediante la pasarela retransmitimos la parte que consideramos segura. Aunque la forma del muro dependerá de cada diseño en particular; por lo general, se forma por medio de dos pasarelas, una interna y otra externa, y por dos filtros, uno entre pasarelas y otro entre la pasarela externa y la red. En este diseño estándar el filtro externo se usa para proteger a la red de ataques, mientras que el filtro interno se usa como protección en caso que la seguridad de la pasarela se vea comprometida. Este sistema de protección parece la solución ideal contra ataques a las redes; sin embargo presenta las siguientes desventajas: su construcción es en hardware, lo que significa gastos tanto en los dispositivos como en el mantenimiento de los mismos; como todas las piezas de hardware, los muros requieren también de software, que deberá ser comprado o diseñado y en cualquier caso deberá ser actualizado constantemente; requiere de una administración especializada, con personal capaz de solucionar las eventuales fallas del sistema y la pérdida de algunas características deseables en un sistema abierto. Las ventajas de usar este tipo de protección son: el evitar que la red quede fuera de servicio por un ataque a alguna de las pasarelas y evitar problemas, si un ataque se realiza a través de nuestra red seremos corresponsales del mismo y en algunos casos se nos considera como sospechosos (recuerde el caso del intruso listo).

No es común utilizar este tipo de dispositivos en ambientes universitarios, en general en este ambiente se prefieren los sistemas abiertos; pero, resulta muy conveniente instalar estos dispositivos para proteger máquinas que se encuentren dentro de la red, pero que deban conservar privacidad en la información. Por ejemplo, con la construcción de un muro se podría conectar la red de servicios escolares de la Facultad de Estudios Superiores Cuautitlán a RedUNAM.

El conectar esta red local a RedUNAM ayudaría a la realización de tareas como la transmisión de las calificaciones de los estudiantes, el tramite de certificado de estudios que tarda un mes podría realizarse en horas, esta información puede transmitirse de forma cifrada para evitar que alguien pueda modificarla. En los ambientes comerciales, muchas compañías estarán dispuestas a afrontar los gastos de un dispositivo de seguridad que les permita garantizar la privacidad de la información. En la figura 6.4 se muestra de forma esquemática el funcionamiento de estos muros, tomando como ejemplo la red Cuautitlán2. Esta red puede dividirse en diferentes sub-dominios: centro de cómputo, edificio de ingeniería, agropecuarias, biblioteca, servicios escolares y edificio de gobierno. En el esquema planteado la conexión a los tres últimos sub-dominios debe estar restringida. Las redes restringidas no tienen conexión más que con la red del centro de cómputo[3]; pero, como ésta se encuentra conectada a las otras sub-redes, la restricción no tiene sentido, para asilarlas de forma efectiva se debe usar un muro. Si se colocan los muros en las posiciones indicadas, las redes uno, dos y tres se pueden comunicar libremente, pero la comunicación de estas, con las redes cuatro, cinco y seis está limitada por medio de un muro.

[3] Esta restricción se puede construir de forma lógica, se configura la red de tal forma que sólo establezca comunicación con ciertas máquinas.

En la actualidad, la sub-red de servicios escolares se encuentra aislada, no forma parte de la red Cuautitlán 2, mediante el uso de un muro adecuado podría integrarse a la red general; con lo que se podría establecer un servicio consultas. Se podría brindar a los usuarios la oportunidad de consultar sus calificaciones, fechas de: inscripción, exámenes ordinarios y extraordinarios, requisitos para algún trámite en especial, etc.

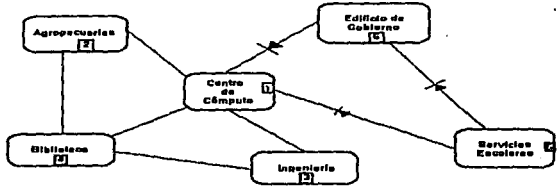


Figura 6.4 Configuración aconsejable para la red Cuautitlán2

Existen tres versiones de estas barreras protectoras: filtrado de paquetes, pasarelas circulares y pasarelas de aplicaciones. El filtrado de paquetes, aunque es la más barata, también es la más ineficiente de estas versiones, consiste en analizar el contenido de los paquetes que pasan a través del muro y en base a su dirección origen y destino saber si el paquete es aceptable o si se rechaza.

El administrador establece una lista de las máquinas desde donde se puede transmitirpaquetes y de las que no deben ser aceptados; e inclusive, se puede especificar además que tienen derechos a usar y a través de que puerto. El defecto de esta versión de los muros es que requiere de buenos conocimientos respecto al manejo de los puertos, un error en la asignación de estos puede abrir el sistema a un ataque. Otro de los defectos de este tipo de muro es que no es capaz de manejar mensajes fragmentados, sólo el primer paquete contiene las direcciones origen y destino, mediante las cuales el muro realiza su trabajo.

Si la única intención del muro es evitar el acceso no autorizado, no presentará ningún problema; al rechazar el paquete inicial el mensaje no puede reconstruirse y el ataque no tiene resultado; pero, si el propósito del muro es el proteger la exportación de información, se necesitará de otro tipo de muro, ya que al no poder determinar si los paquetes son aceptables o no los dejará pasar. Además esta versión del muro tendrá problemas con servicios como FTP y las tracciones X11. El problema con FTP se debe a que la asignación de puertos es aleatoria (se toma el que este disponible), y por lo tanto un intruso puede obtener acceso a la máquina mediante una conexión FTP y uno de los puertos superiores (que generalmente no son usados por FTP), para defenderse de este tipo de ataque se podría modificar el código de la función FTP y realizar la asignación de puertos de forma pasiva; sin embargo, no todos las máquinas admiten esta modificación. El problema con las transacciones X11 es similar al encontrado con FTP, se necesita de una llamada de entrada para realizar el trabajo de esta manera un intruso puede utilizar esta puerta para accesar al sistema, para evitar problemas con X11 es recomendable limitar el número de puertos a través de los cuales se realizan las transacciones.

Las pasarelas de aplicaciones son la versión más efectiva de los muros, en lugar de usar un mecanismo de propósito general para permitir el paso de diferente tipo de señales se usa un código de propósito especial para cada aplicación deseada. De esta manera, no debemos preocuparnos acerca de la interacción de diferentes entidades, ni acerca de los errores de seguridad en cientos de máquinas que ofrecen servicios (que deberían ser seguros, pero no lo son) a los usuarios. Sólo necesitamos elegir un par de programas para implementar un mecanismo de seguridad de este tipo. La mayor ventaja de esta versión de los muros es la facilidad que brindan para el control del tráfico de entrada y salida de la red.

Los servicios se restringen a ser usados por determinados usuarios, a los cuales también se les asigna un uso limitado del sistema; de esta forma, aunque algún intruso pueda accesar a la red, al no tener asignados recursos físicos no podrá hacer uso de ella. Por lo general, independientemente de la tecnología usada para el resto del muro, el correo electrónico se hace pasar por un muro de pasarela de aplicación, debido a que puede ser un punto de ataque (no se olvide que el gusano de Internet usa el correo electrónico para efectuar su ataque). Por lo general, la pasarela de aplicaciones se usa en conjunción con otra forma de muro; por ejemplo, puede usarse junto con un muro de filtrado de paquetes para mejorar las deficiencias de este en lo que respecta a las transacciones X11. La principal desventaja de las pasarelas de aplicación es que requieren un programa especializado o la modificación de la interfase de la mayoría de los servicios prestados, para ser construidas. En la práctica esto significa que sólo se soportarán los servicios más importantes, de allí su necesidad de ser construido en conjunción con un muro de otra versión.

Las pasarelas circulares son otro tipo de muro, mediante este mecanismo el emisor se conecta a un puerto de la pasarela mediante una conexión TCP, la pasarela a su vez se encarga de establecer contacto con el receptor, durante la llamada la pasarela se comporta como una cable interconectado al emisor con el receptor. La conexión al receptor puede hacerse de forma automática o bien puede esperar que el emisor indique el destino, en este caso se necesitará un protocolo para que la pasarela intercambie información con la entidad que solicita la llamada. Mientras la información pasa a través de la pasarela, esta registra el número de bytes y su destino; el análisis de esta información puede mostrar cuando una cuenta ha sido rota.

Como medidas de control este sistema limita el tiempo y la forma de conexión a los puertos, establece un tiempo de retardo antes que los puertos puedan ser rehusados, usa una lista de llamadas válidas a un puerto y requiere que el usuario se identifique para poder realizar el servicio. Todas estas medidas son configurables, dependerán de las necesidades del sistema. El mayor problema con esta versión de los muros es que los programas cliente deben modificarse para que puedan interactuar con el muro; sin embargo, estos cambios en general no serán muy difíciles. Los muros son una herramienta bastante poderosa para establecer la seguridad de una red; pero, también tienen sus limitaciones, las cuales es necesario entender. El muro es un buena defensa contra ataques planteados en los niveles más bajos del modelos de protocolos; pero, su protección en contra de ataques en las capas superiores es limitada o inclusive nula, de allí sus problemas para trabajar con aplicaciones como el X11, la única forma de asegurarse en contra de estos ataques es haciendo que el muro se niegue a ejecutar estos programas de aplicación.

Por otro lado, el grado de protección de un muro dependerá de la calidad del código que se encarga de establecer los permisos de paso. De esta forma, una pasarela de correo electrónico (pasarela de aplicación) debe ser sumamente cuidadosa con la forma en que utiliza los diferentes protocolos de correo electrónico y debe usar todas las funciones de entrega de mensajes correctamente, de lo contrario la misma aplicación generará problemas. El problema generado por el error en la aplicación puede estropear el enlace entre la pasarela y la red, lo que puede ser interpretado como un ataque. A pesar de sus problemas, los muros tienden a convertirse en la forma de defensa más poderosa; sin embargo, tampoco debe olvidarse que la lucha por tener un sistema seguro es constante, no podemos crear una forma de defensa perfecta, como tampoco podemos lograr una forma de ataque perfecta. Nuestro trabajo como administradores será hacer la tarea del intruso lo más difícil posible.

Conclusiones

La principal ventaja del uso de las redes de computadoras, es que estas brindan a los usuarios enormes cantidades de información y con una amplia gama de recursos computacionales; sin embargo, las redes son sistemas complejos, que requieren de trabajo y dedicación para su construcción mantenimiento y coordinación. El trabajo del administrador es evitar que el usuario final tenga que lidiar con la complejidad del sistema. Otra de las tareas fundamentales del administrador de una red, y sobre todo si se trata de una red dentro de un ambiente académico, como la red de la Facultad de Estudios Superiores Cuautitlán, es el promover el uso de la misma.

Las redes están cambiando muchas cosas, no sólo brindan una nueva forma de comunicación, sino que crean la necesidad de estas formas de comunicación, las empresas si desean ser competitivas tienen que negociar de forma global y la red es el vehículo ideal para hacerlo. Poco a poco las redes modifican la forma en que nos relacionamos, dentro de poco el mismo ejercicio profesional será diferente, quizá no estemos tan lejos del día donde se puedan realizar labores de forma remota. La Universidad Nacional Autónoma de México, consciente de esto participa en la red Internet a través de la RedUNAM, la mayoría de las facultades e institutos están enlazados y tienen acceso a Internet. El siguiente paso ahora es difundir el uso de esta maravillosa fuente de comunicación; pero para promover este sistema es necesario que las redes locales sean confiables, de lo contrario nadie querrá usarlos. Además, es necesario tener en cuenta que no todos los usuarios serán expertos en el uso de sistemas de cómputo por lo tanto, es conveniente presentar a ellos el uso de la red mediante una interfase amigable y fácil de usar.

Tanto la confiabilidad del sistema, como su promoción necesitan de una buena administración de la red. La administración debe ser llevada a cabo por gente dispuesta superarse día a día, porque los sistemas de cómputo avanzan a velocidades agigantadas. Es necesario de un grupo interdisciplinario con conocimientos tanto de hardware como de software, así como gente preparada en materia de comunicaciones, para poder crear un sistema de red confiable y funcional. El ambiente de la Facultad de Estudios Superiores Cuautitlán es sumamente propicio para el desarrollo de un sistema con estas características, se cuenta con estudiantes de Ingeniería con conocimientos en hardware y en comunicaciones y con estudiantes de informática que podrían cubrir lo que se refiere a manejo de software y servicios. Pero no sólo eso, Internet ofrece una multitud de servicios que cualquier persona, sin importar su campo de trabajo, puede explotar (parte de la información contenida e este trabajo fue obtenida de la red).

En la actualidad uno de los aspectos más importantes de la administración de una red es la seguridad. La popularidad de Internet ha generado su expansión hacia múltiples campos, con lo cual también se generó la necesidad de un sistema de protección. El énfasis en los aspectos de seguridad no es un capricho, debemos estar conscientes del valor de nuestra información y sobre todo del valor de la información de terceras personas, debemos cuidar que nuestra información no pueda ser dañada y a la vez cuidar que nuestro sistema no sirva como puerta de ataque.

La seguridad de una red se da en dos campos: el primer aspecto que contempla la seguridad es evitar el acceso de personas no autorizadas, esta filosofía de seguridad supone que el peligro proviene de fuera de la red, de esta forma si se evita el acceso a personas ajenas al sistema, éste se encontrará seguro.

Por otro lado, no podemos construir un sistema de seguridad infalible (como ya hemos señalado), además debemos considerar también como un factor de riesgo a los mismos usuarios de la red, se debe garantizar que la información contenida en el sistema no pueda ser atacada desde dentro del mismo; esta es la segunda forma de seguridad, cuidar el sistema desde dentro. Nosotros consideramos que una buena política de seguridad debe cuidar ambos aspectos.

Para la red de la Facultad de Estudios Superiores Cuautitlán recomendamos que se utilice el sistema *npasswd* como medida de seguridad contra ataques externos. Este sistema se encarga de establecer un sistema de contraseñas más estricto, recuerde que uno de los principales problemas de seguridad en las redes se generan por la debilidad en las contraseñas, este sistema cubriría la primera parte del sistema de seguridad.

Para proteger el sistema desde dentro recomendamos el uso de un sistema de identificación (authentication) como *kerberos*. Este sistema supone que el intruso puede leer el tráfico de la red, para evitar que pueda hacer uso de la información que viaja a través de la red la información se transmite cifrada. Por lo tanto, aunque pueda leerla no puede utilizarla.

Consideramos que una configuración de este tipo sería bastante fuerte; sin embargo teniendo presente que no existen medidas infalibles recomendamos que periódicamente se realicen auditorías al sistema, con herramientas como *COPS* para detectar los puntos débiles de la red y corregirlos. Es necesario que el equipo de administración de la red esté en constante contacto con los usuarios, que les recomiende fuentes confiables para obtener información, que sepa el software que utilizan e instalan.

Además es muy recomendable establecer un sistema de contabilidad de los recursos, por un lado para revisar el rendimiento del sistema y para evitar que alguien abuse de los recursos (recuerde que el caso del intruso astuto empezó por una diferencia de centavos en la contabilidad del sistema).

Por último debemos recordar a los futuros administradores que todos los aspectos del control de una red cambian día a día, por lo que se requiere de disciplina y dedicación para administrar adecuadamente una red. Se debe estar muy pendiente de la información que se publica a través de Internet para descubrir nuevas aplicaciones, nuevos sistemas de seguridad o defectos en los ya existentes, etc. El administrar una red es un reto que debe ser atacado de forma disciplinada y continua. Por ejemplo durante el mes de junio de 1995, se publico un artículo en la revista Unix Review, en el cual se habla de un nuevo protocolo para Internet, el cual ampliará las prespectivas de la red ya que al ser un protocolo de 128 bits en lugar del actual que es de 32. Esto demuestra que las impresionantes aplicaciones que podemos encontrar en la red actualmente son sólo el principio.

Bibliografía

Stephen J. Kochan / Patrick Wood.- "Exploring the Unix System"

SAMS

David A. Curry.- "Unix System Security"

Addison Wesley

Stephen Coffin.- "Unix (Sistema V versión 4)"

McGraw Hill

Rick Farrow.- "Unix System Security"

Addison Wesley

Evi Nemeth / Gart Snyder / Scott Seebas.- "Unix System Administration Handbook"

Prentice Hall

Hilbert Held.- "Understanting Data Communications"

John Wiley & Sons

Andrew S. Tandembaum.- "Redes de ordenadores"

Prentice Hall

Michael A. Miller.- "Introduction to Digital Communications"

WEST

Uyless Black.- "TCP/IP and related Protocols"

McGraw Hill

Gary C. Kessler / David A. Train.- "Metropolitan Area Networks"

McGraw Hill

William R. Cheswick / Steven M. Bellovin.- "Firewalls and Internet Security"

Addison Wesley

D. Russell.- "The Principles of Computer Networking"

Cambridge Universitary Press

Gerd E. Keiser.- "Local Area Networks"

McGraw Hill